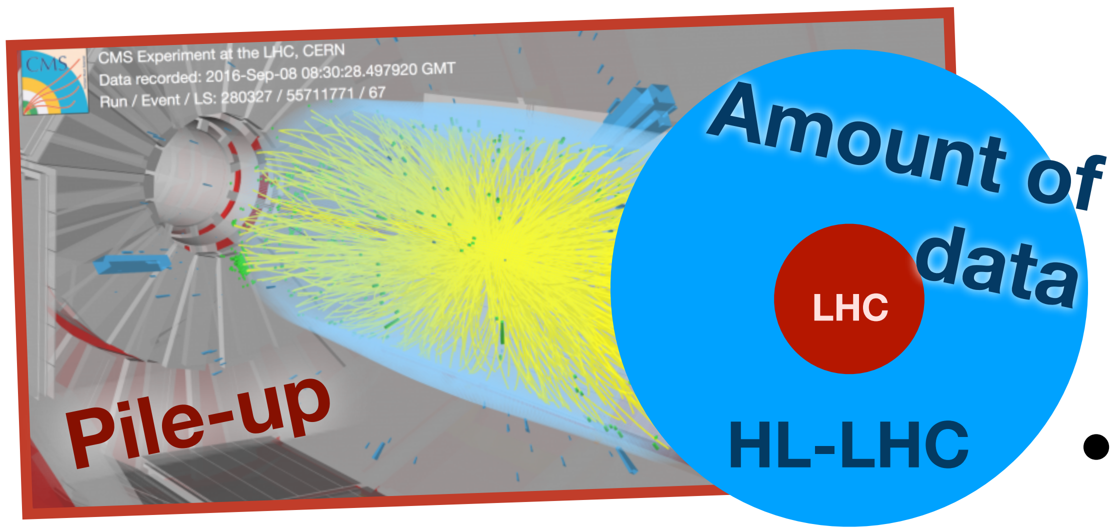
HGCAL reconstruction using deep learning

Thomas Klijnsma

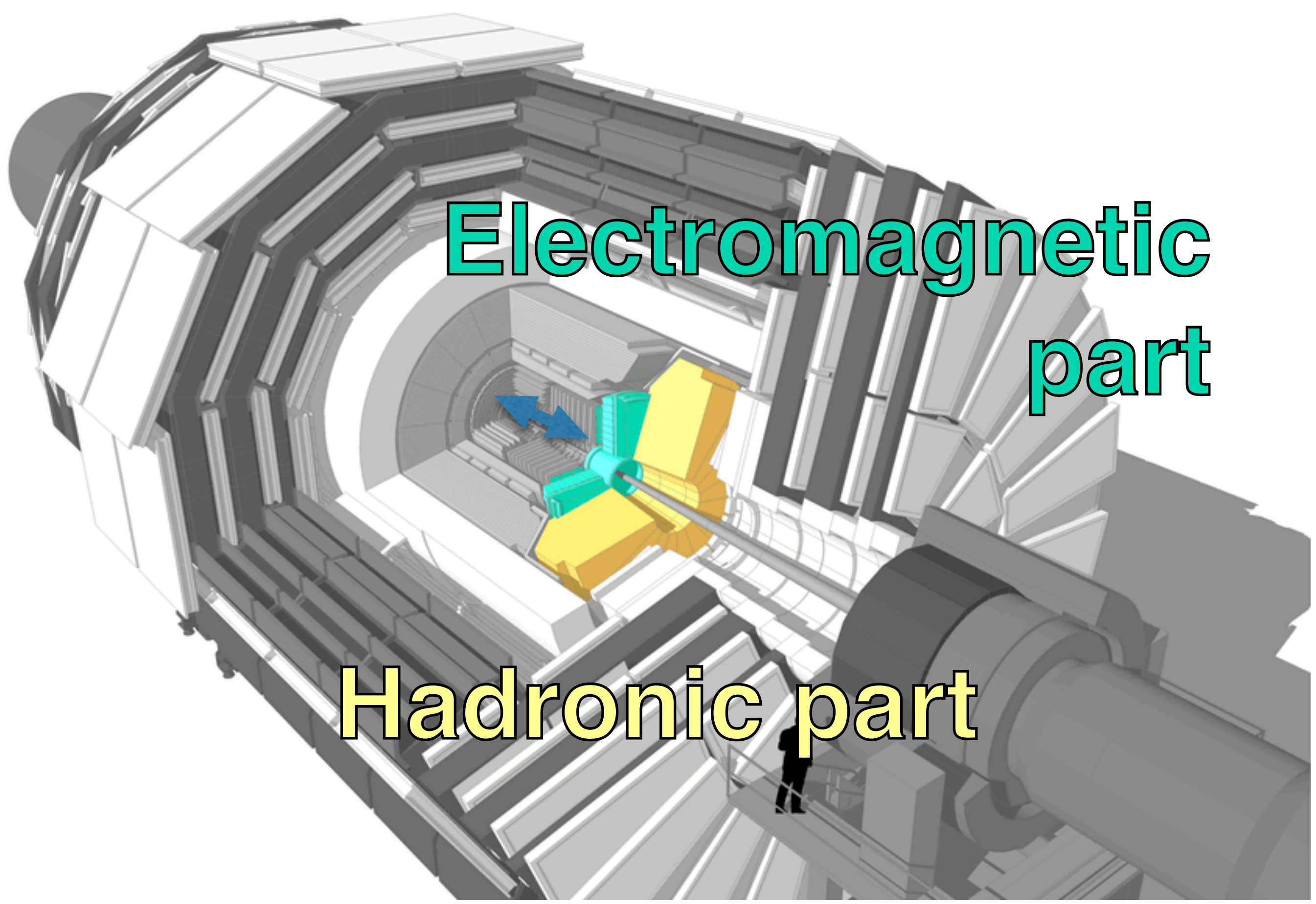
3 February 2021

Fermilab

ML @ LHC

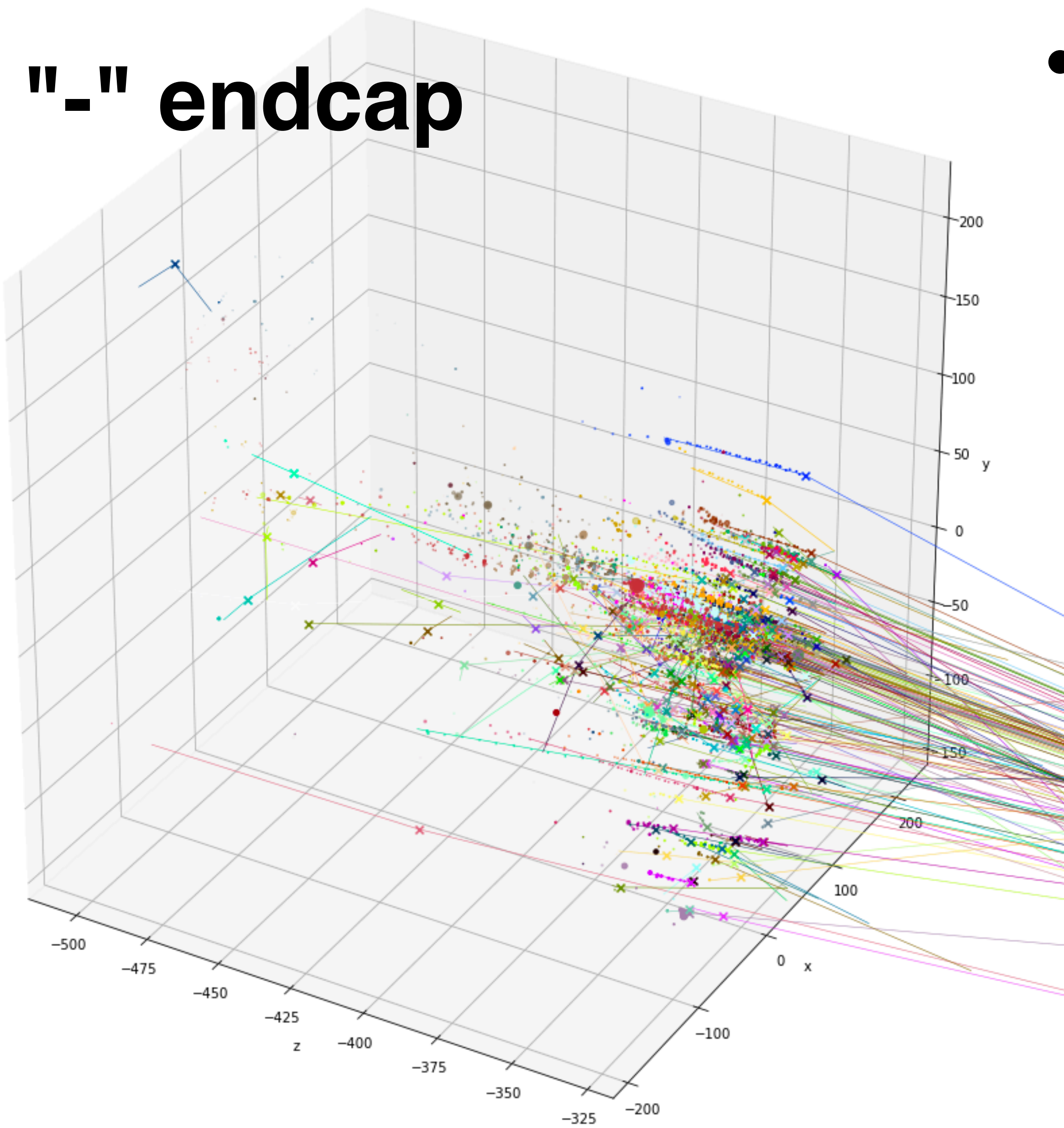


- Most traditional clustering algorithms scale **combinatorially**
  - **Explodes** going from 32 to **200** PU interactions
- ML could provide a **constant time clustering**
  - HGCAL is placed at both endcaps of the CMS detector
  - Sampling calorimeter: raw data is a **3D point cloud + timing and energy** (i.e. 5D)
  - Along beamline; particularly busy environment

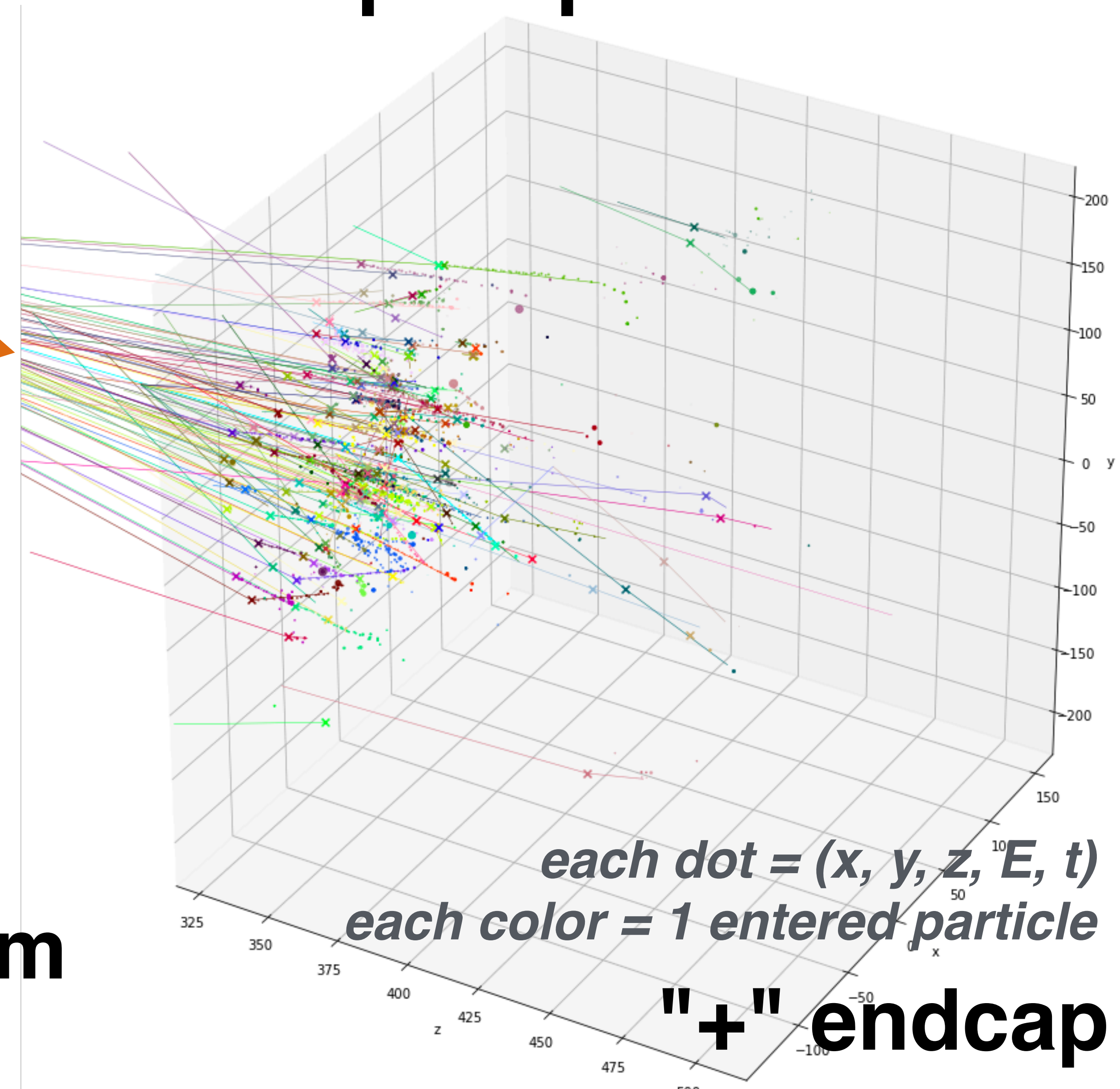


Example event display: ttbar @ 13 TeV

"-" endcap



- This is just truth-level:
  - Real events will a huge amount of **noise**, and a huge amount of **pileup**

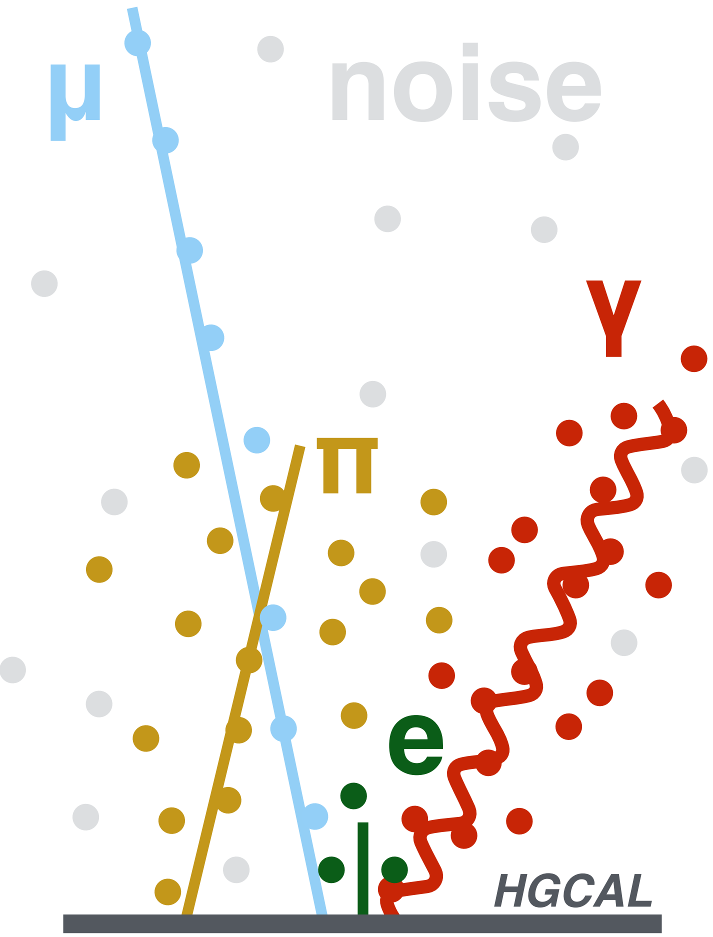


each dot = (x, y, z, E, t)
each color = 1 entered particle

"+" endcap

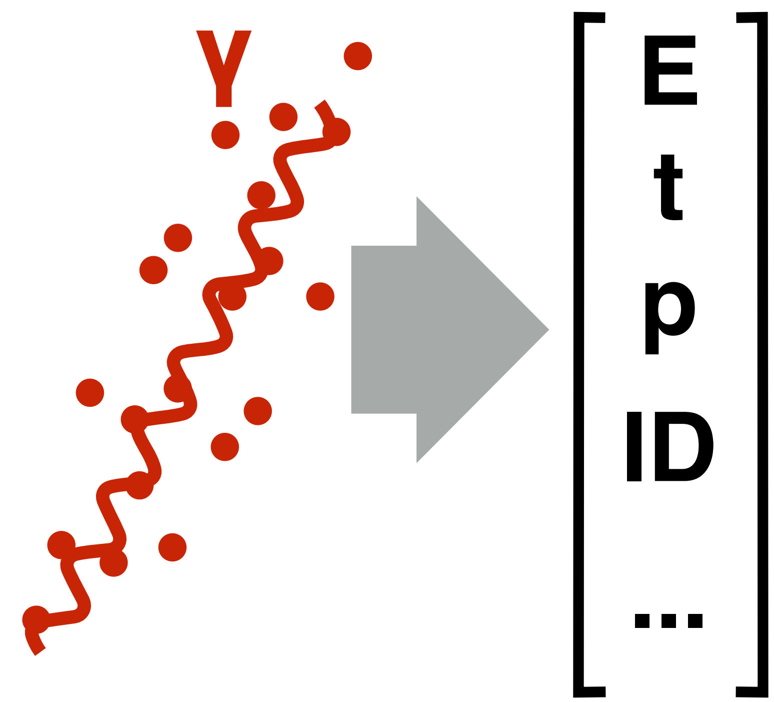
- Challenges reconstructing entered particles: **Can we get back this clustering back from an unmarked point cloud?**

Clustering vs. Property determination



Clustering

Goal: Correctly assign hits to the right particle



Property determination

Goal: High accuracy determination of particle properties

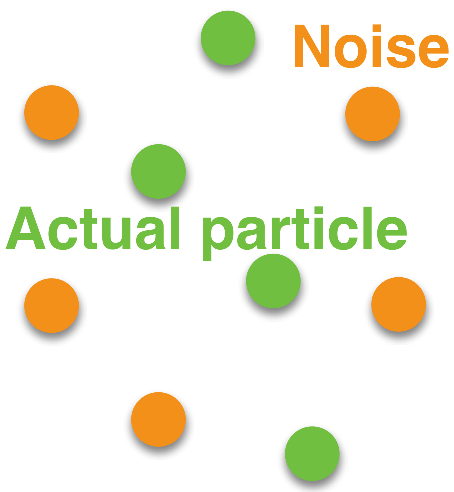
GNNs
DGCNN/EdgeNet
GravNet/GarNet

Point-cloud based NNs
PVCNN

GNNs
Dynamic Reduction Network

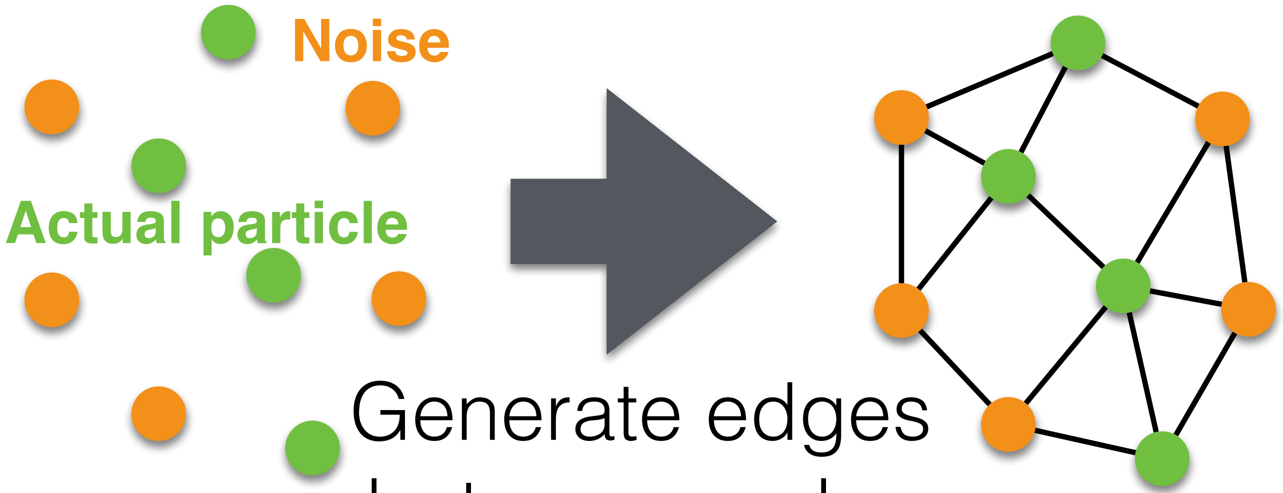
One-shot approaches
Object condensation

A very simple GNN for clustering



Cloud of HGCAL hits (x, y, z, E, t)

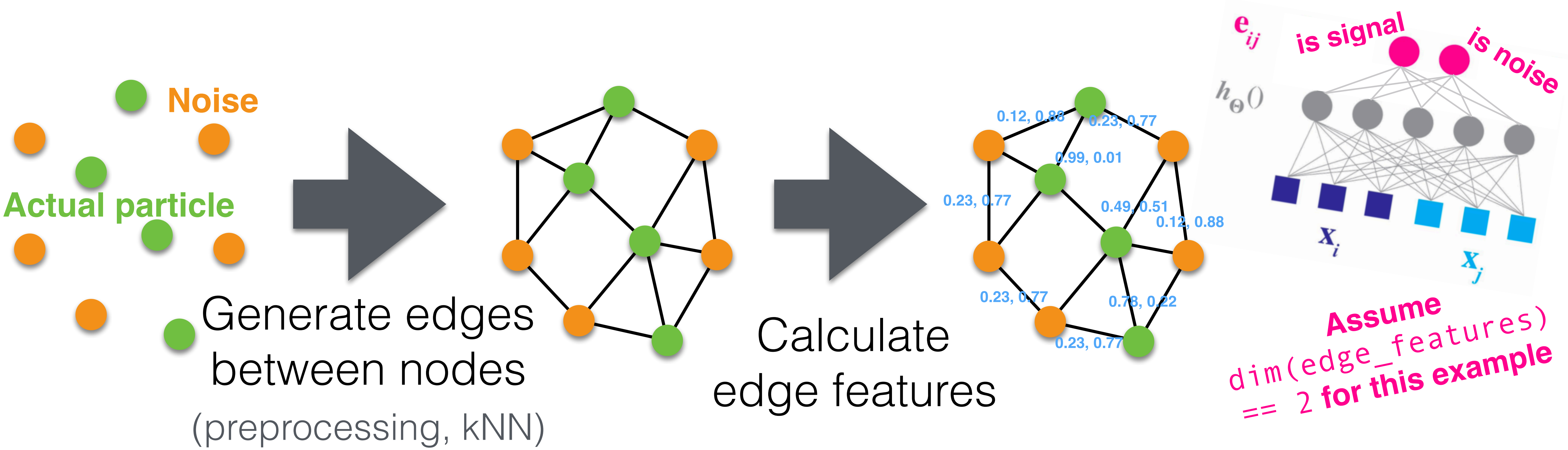
A very simple GNN for clustering



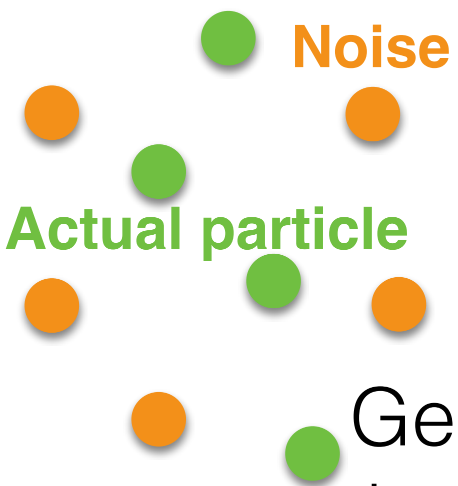
Generate edges between nodes

(preprocessing, kNN)

A very simple GNN for clustering

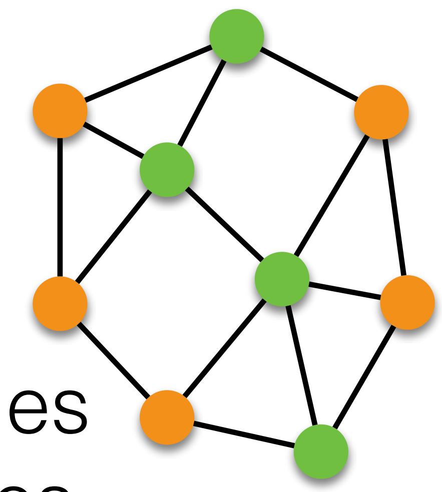


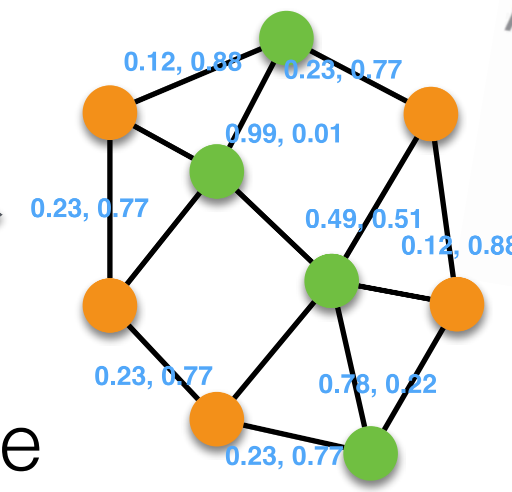
A very simple GNN for clustering



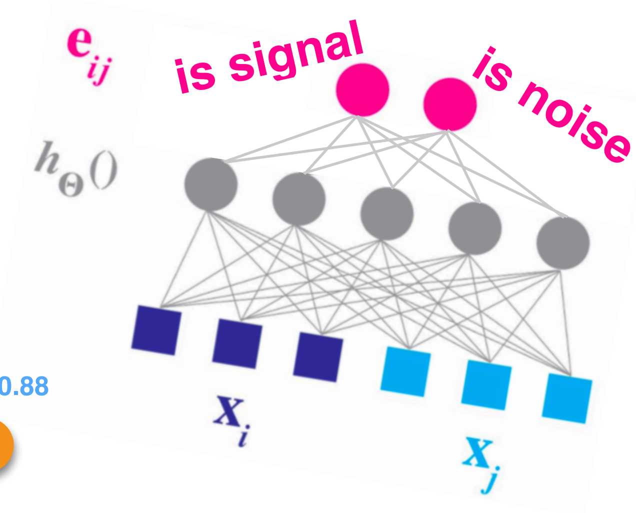
Generate edges between nodes

(preprocessing, kNN)

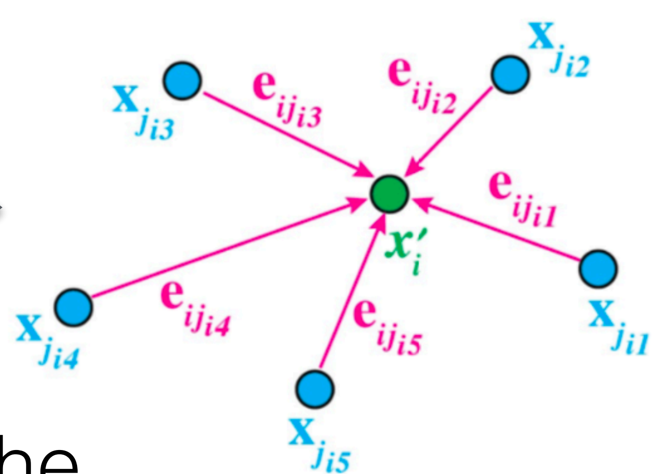




Calculate edge features



Assume `dim(edge_features) == 2` **for this example**



Update the node features using the edge features

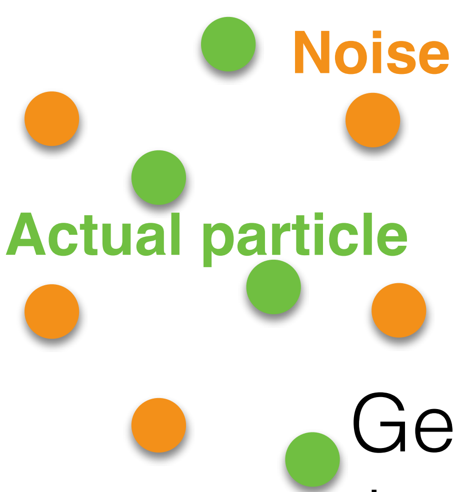
Also called graph convolution

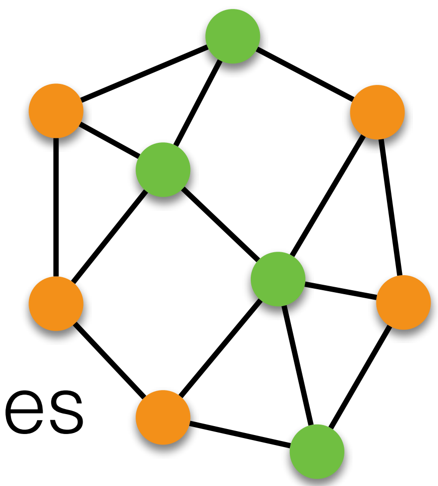
Can envision a 'sliding window' (window size = k nearest neighbours) of neighbors influencing a point

'Message-passing' takes place

Important information from a neighbor spreads via x_i to another neighbor in the next iteration

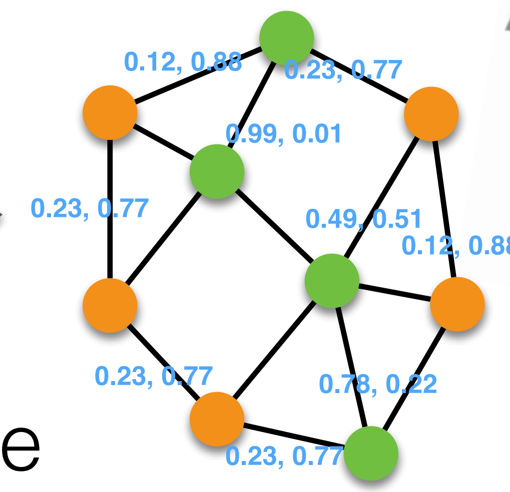
A very simple GNN for clustering



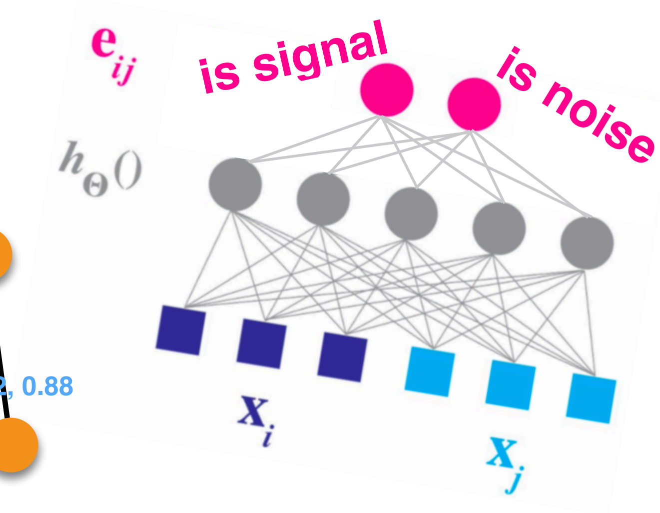


Generate edges between nodes

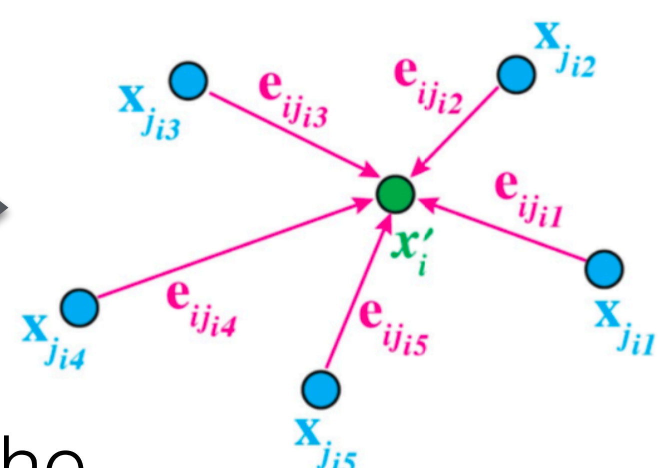
(preprocessing, kNN)



Calculate edge features

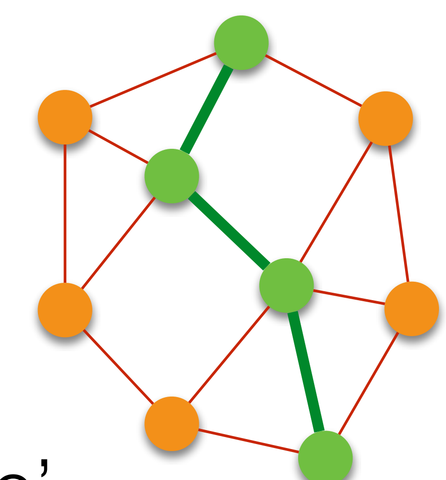


Assume `dim(edge_features) == 2` **for this example**



Update the node features using the edge features

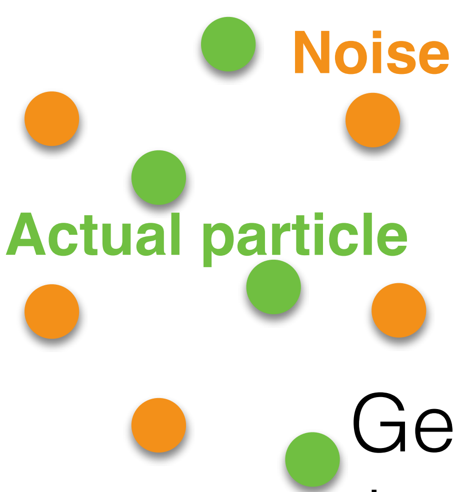
(repeat)



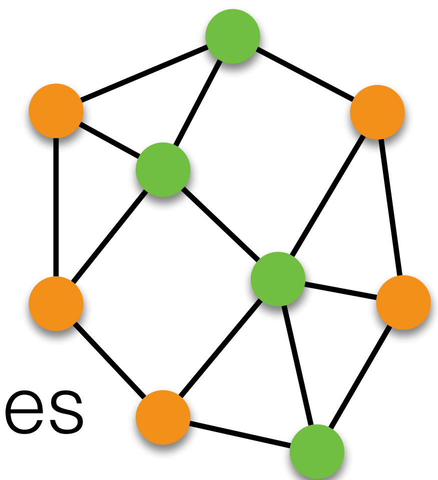
Classify into 'true' edges vs 'false' edges

(ML, 2 edge features)

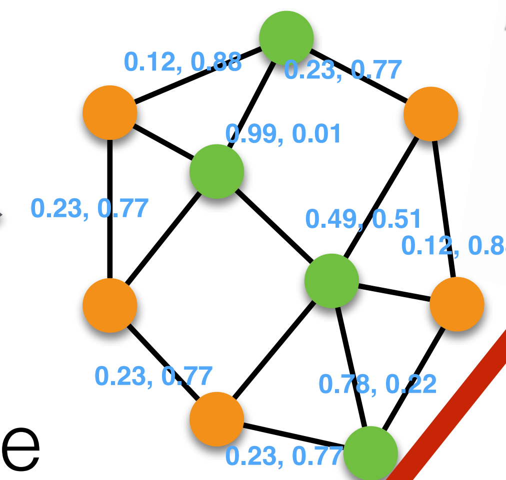
A very simple GNN for clustering

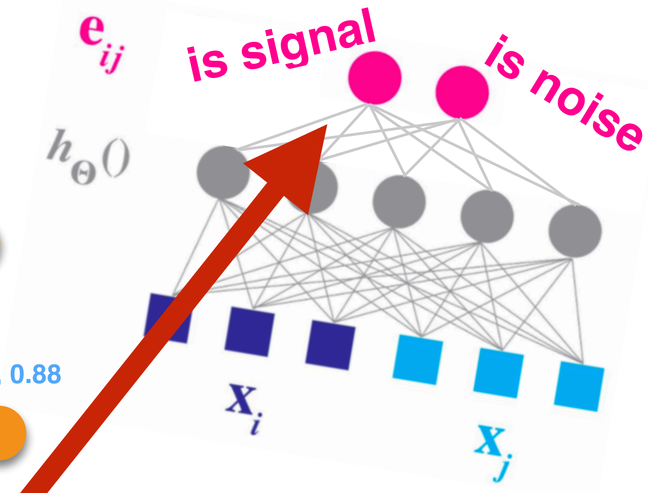


Generate edges between nodes
(preprocessing, kNN)

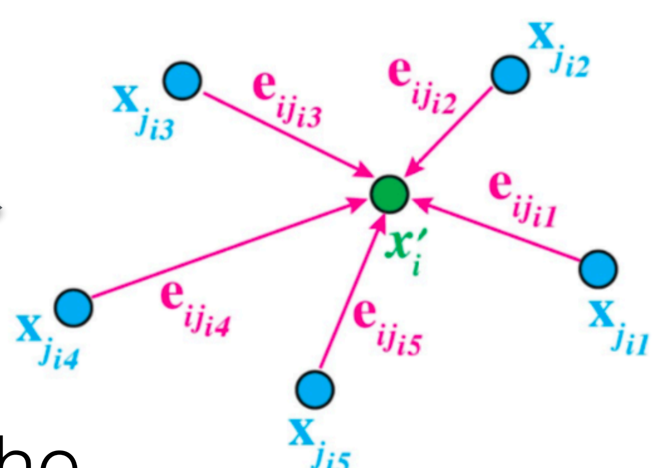


Calculate edge features



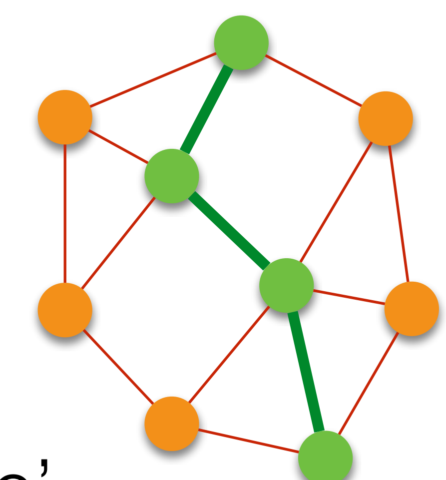


Assume `dim(edge_features) == 2` **for this example**



Update the node features using the edge features

(repeat)

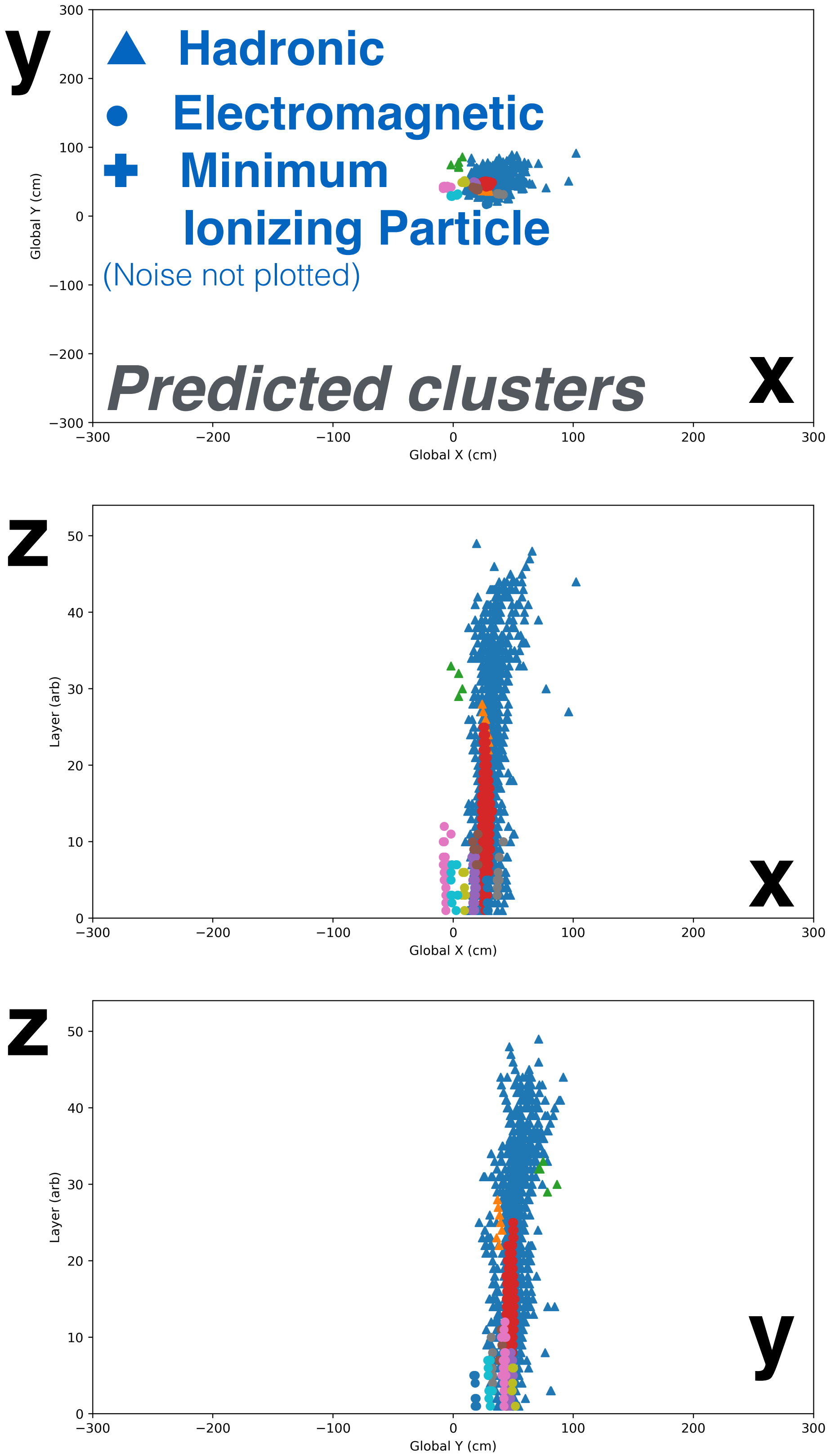


Classify into ‘true’ edges vs ‘false’ edges
(ML, 2 edge features)

Can classify into arbitrary number of categories too

GNN success: Hadronic τ decay

*Skipping **single-particle zero PU** results; they look great and are not considered a challenge anymore*

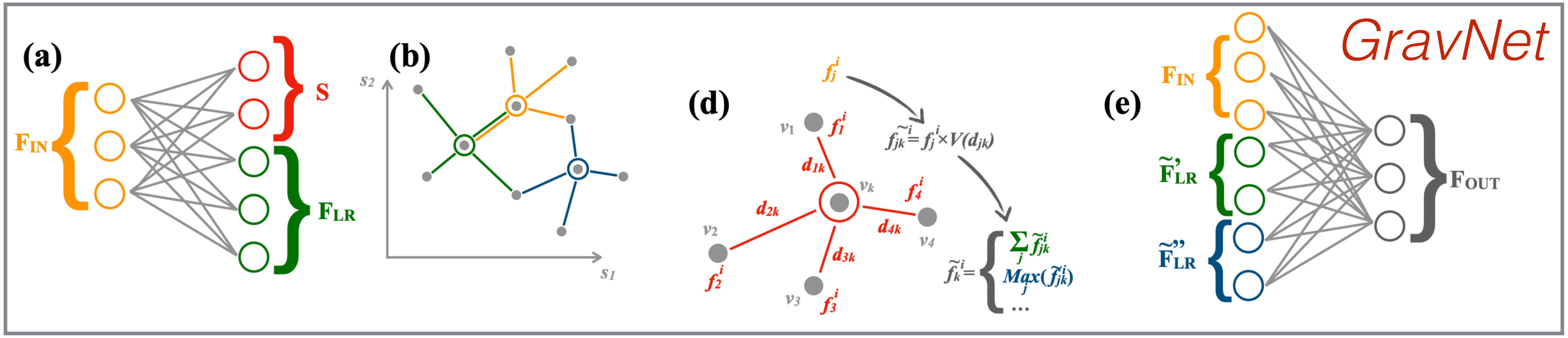


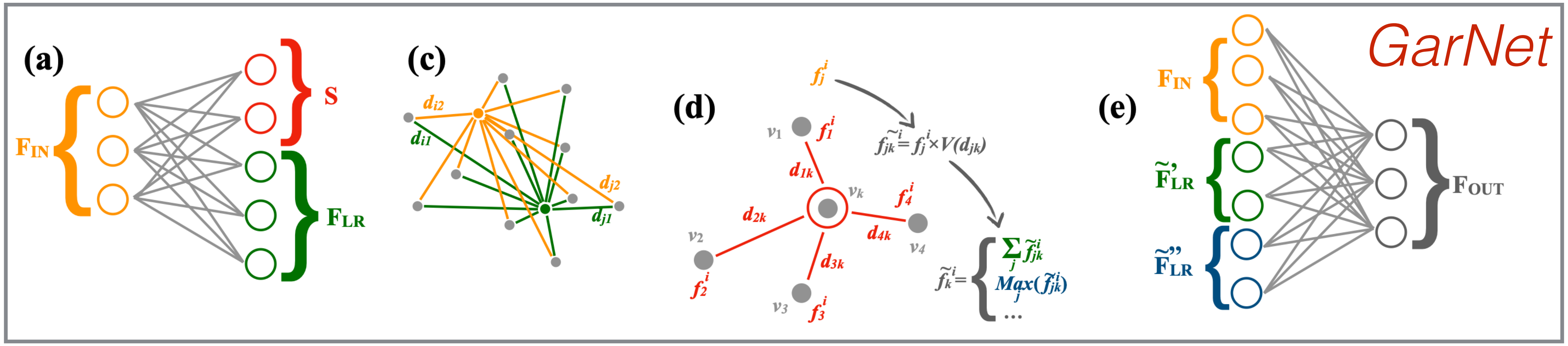
- **Different colors = different particles**
  Clear particle-like clusters are constructed
- Architecture using **EdgeConv** layers (1801.07829)
- Used simple union-finding algorithm for instance segmentation, to be improved
- Being implemented for reconstruction in **HCAL** *by Jeff Krupa (MIT) et al.*

GravNet/GarNet

[1902.07987]

- **DGCNN/EdgeNet** uses *large* amount of memory and keeping *inference time* under control is a challenge
- **GravNet/GarNet** greatly reduces computational needs
  - Split coordinate and feature space
- Large downside: **Number of clusters** is *not* learned (i.e. it's input)

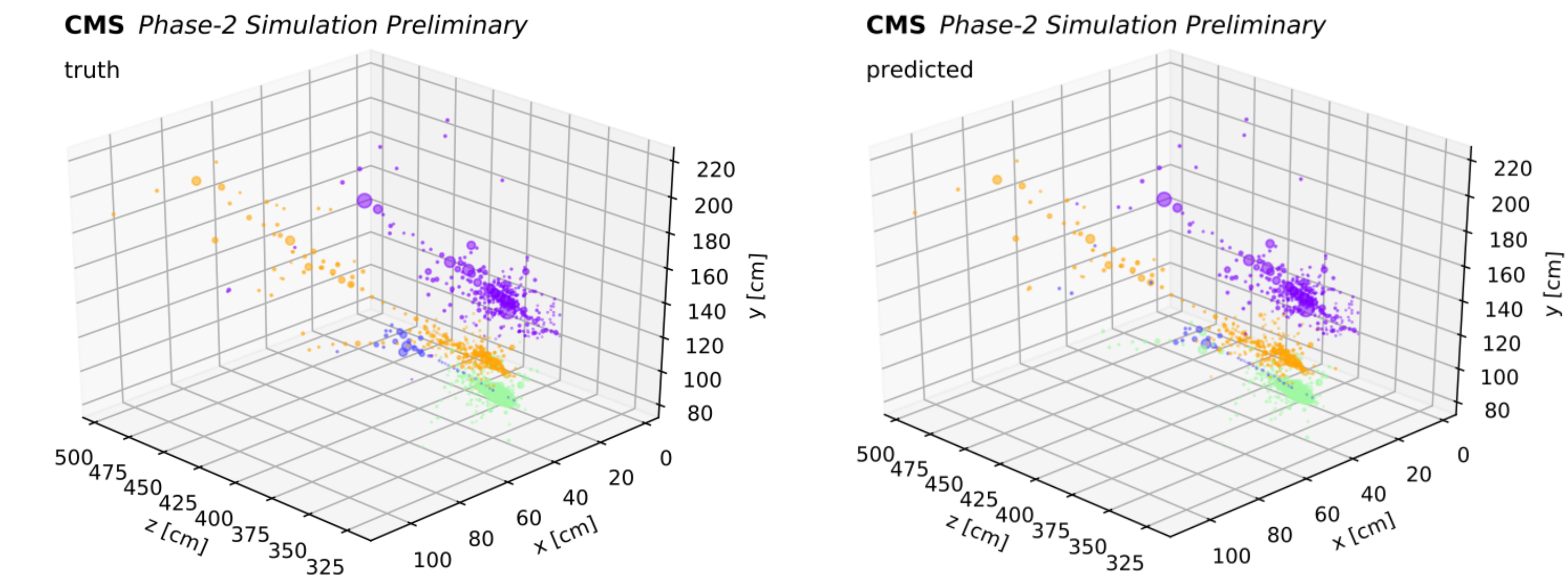


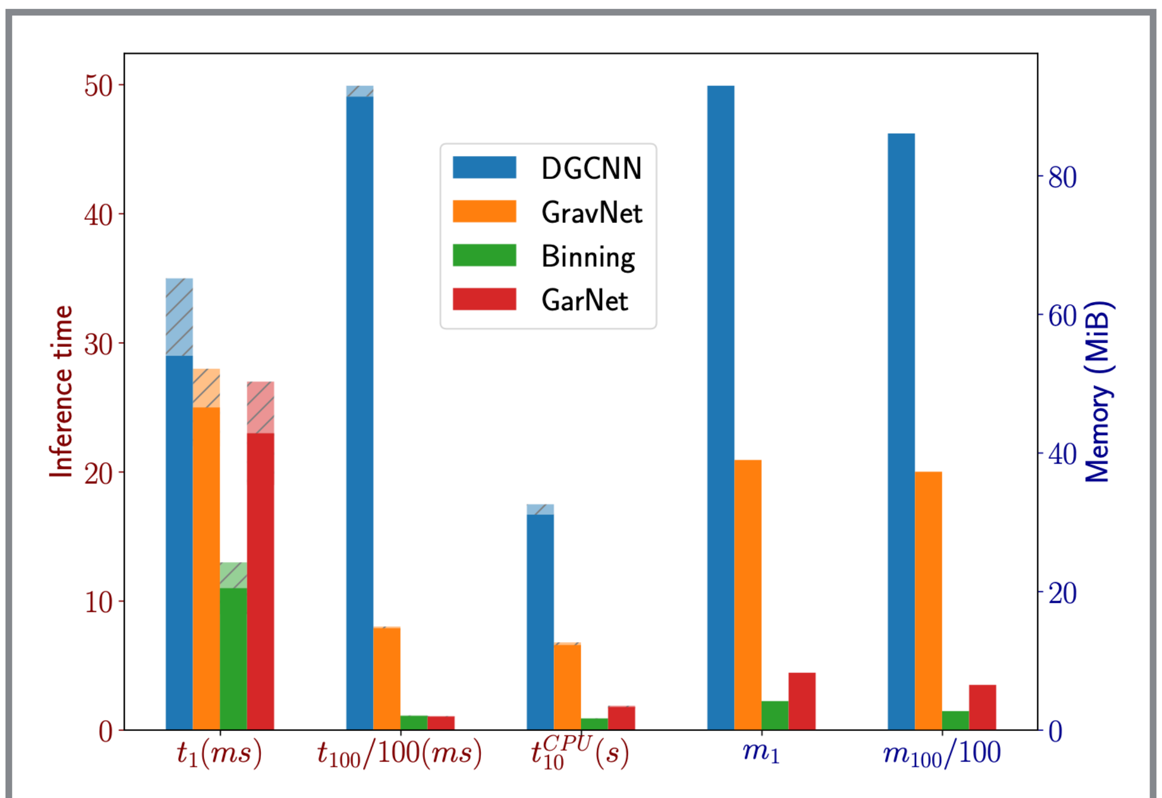


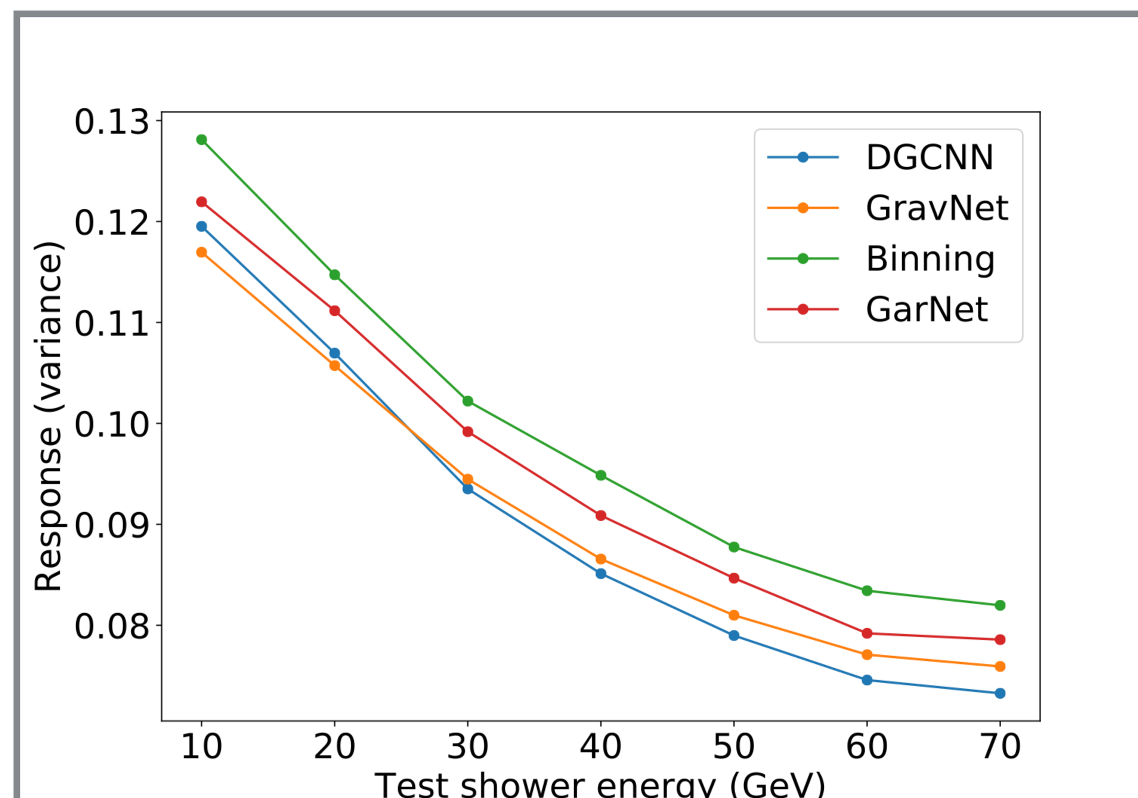
GravNet/GarNet

[1902.07987]

- D … ime
  u
- G
  -
- L …)



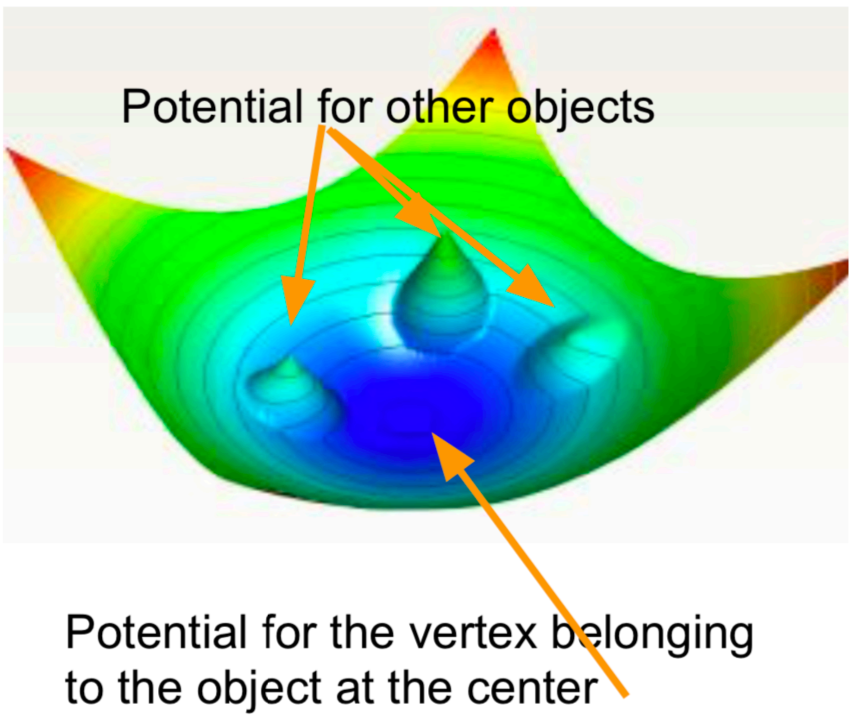


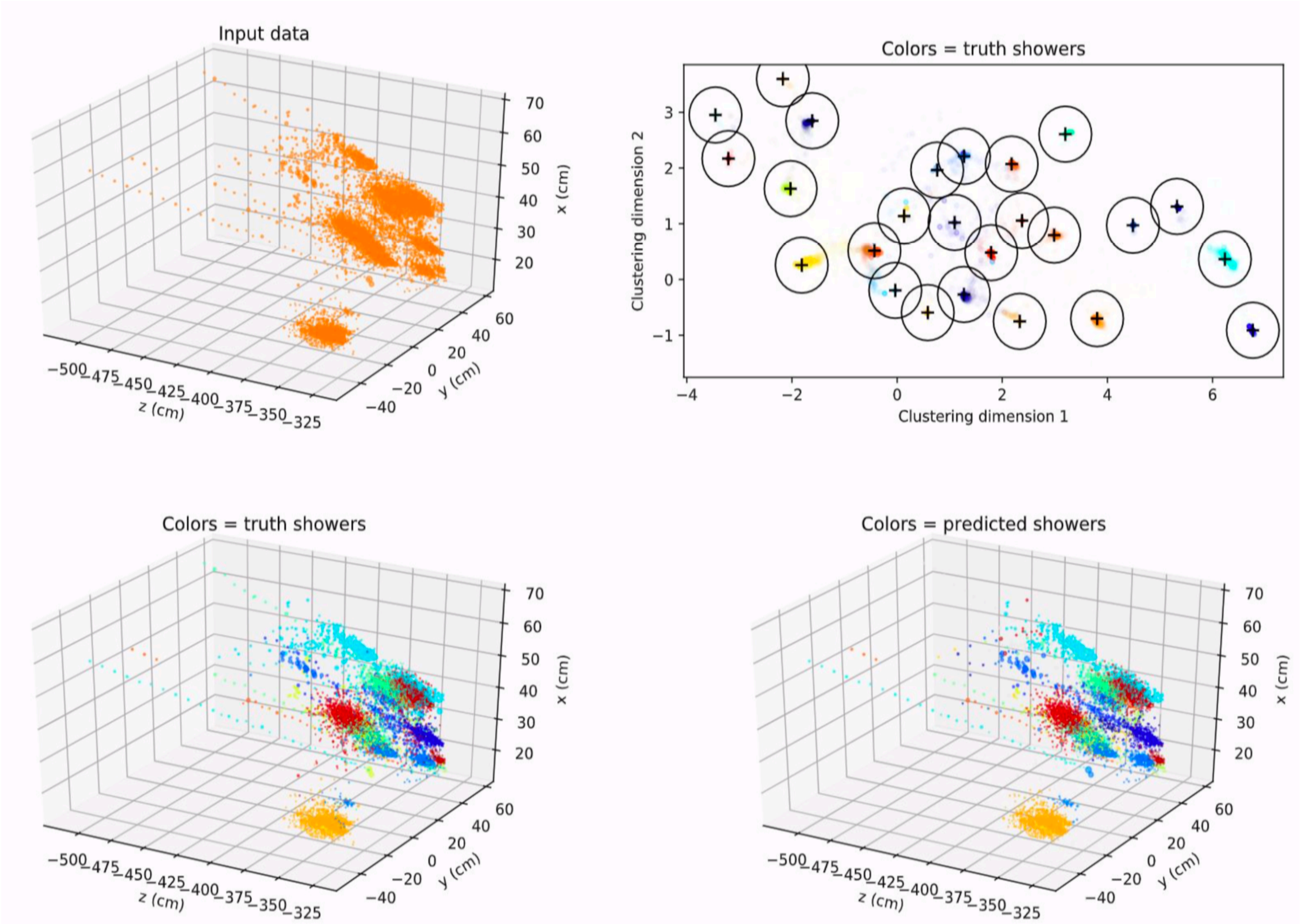


Object condensation

[2002.03605]

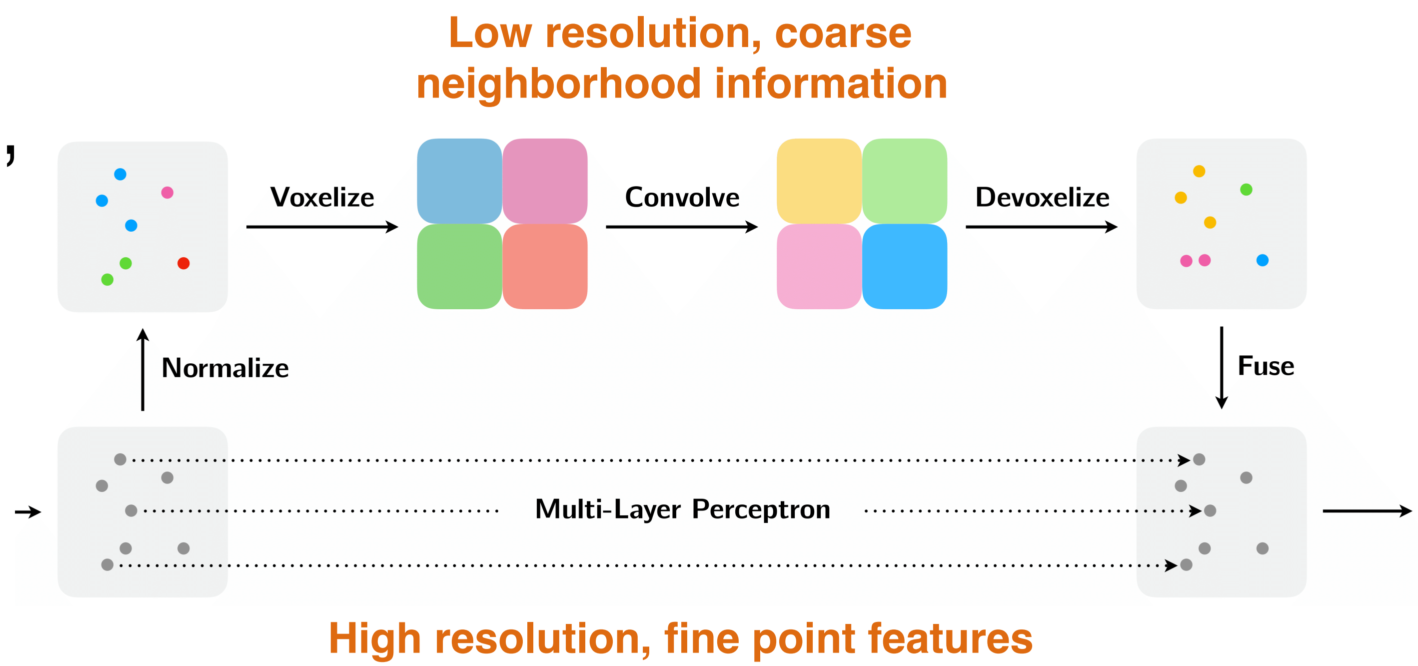
- **One-shot** segmentation + property determination
- **Input** is a set of related pixels/points/vertices/edges/...
  Output is a **number of objects** (e.g. number of particles in an event) each carrying their high-level object properties (e.g. their four-momenta)

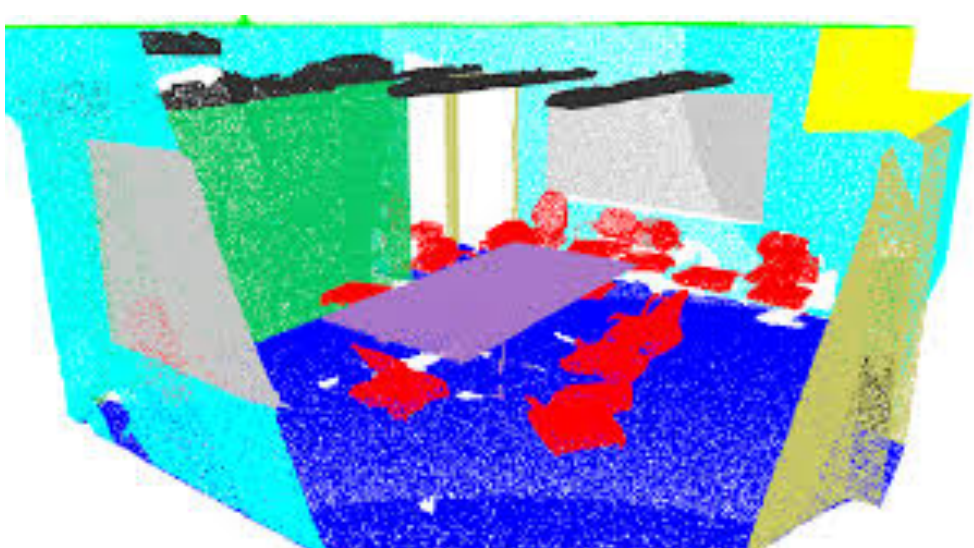




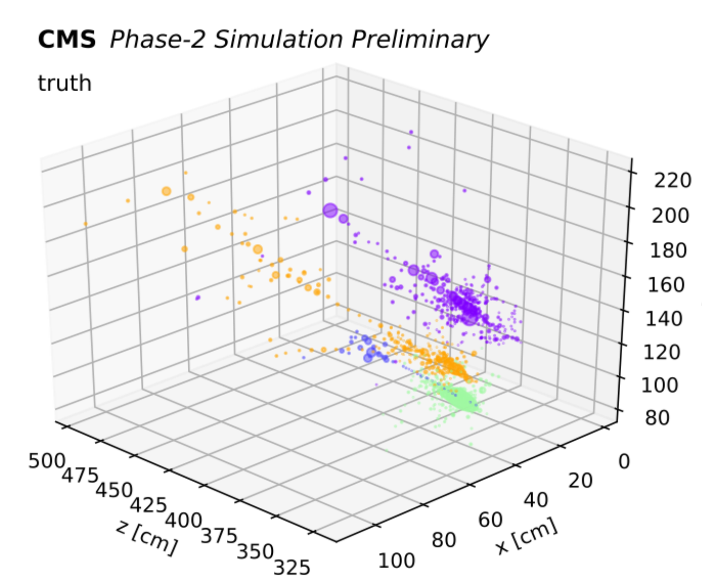
Point-cloud methods: PVCNN

- Alternative to graph-based methods: **Point-voxel CNN** , a point-cloud-based neural network
- **Voxelized convolution** is very **memory efficient** compared to EdgeConv / point-cloud convolution methods





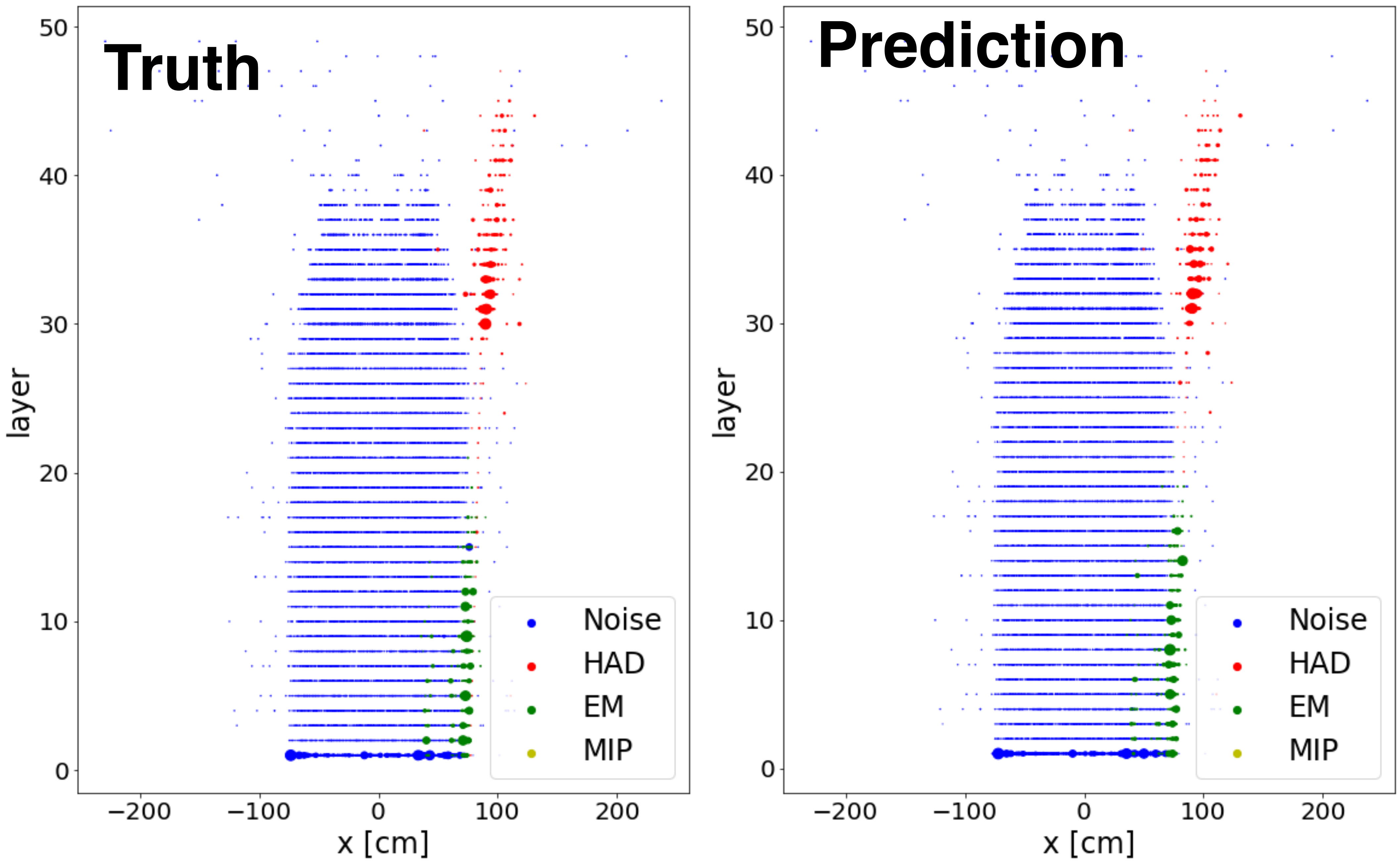
X: (x, y, z, r, g, b)
y: furniture type



X: (x, y, z, E, t)
y: particle type

y:
Hadronic,
EM,
MIP,
Noise

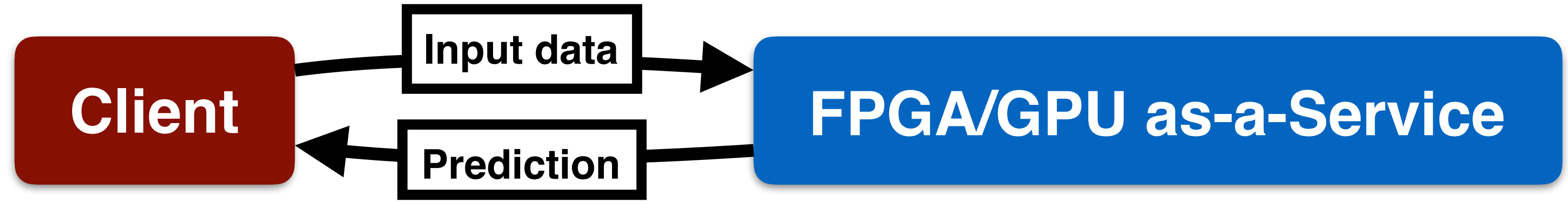
τ event display (single particle + no PU)



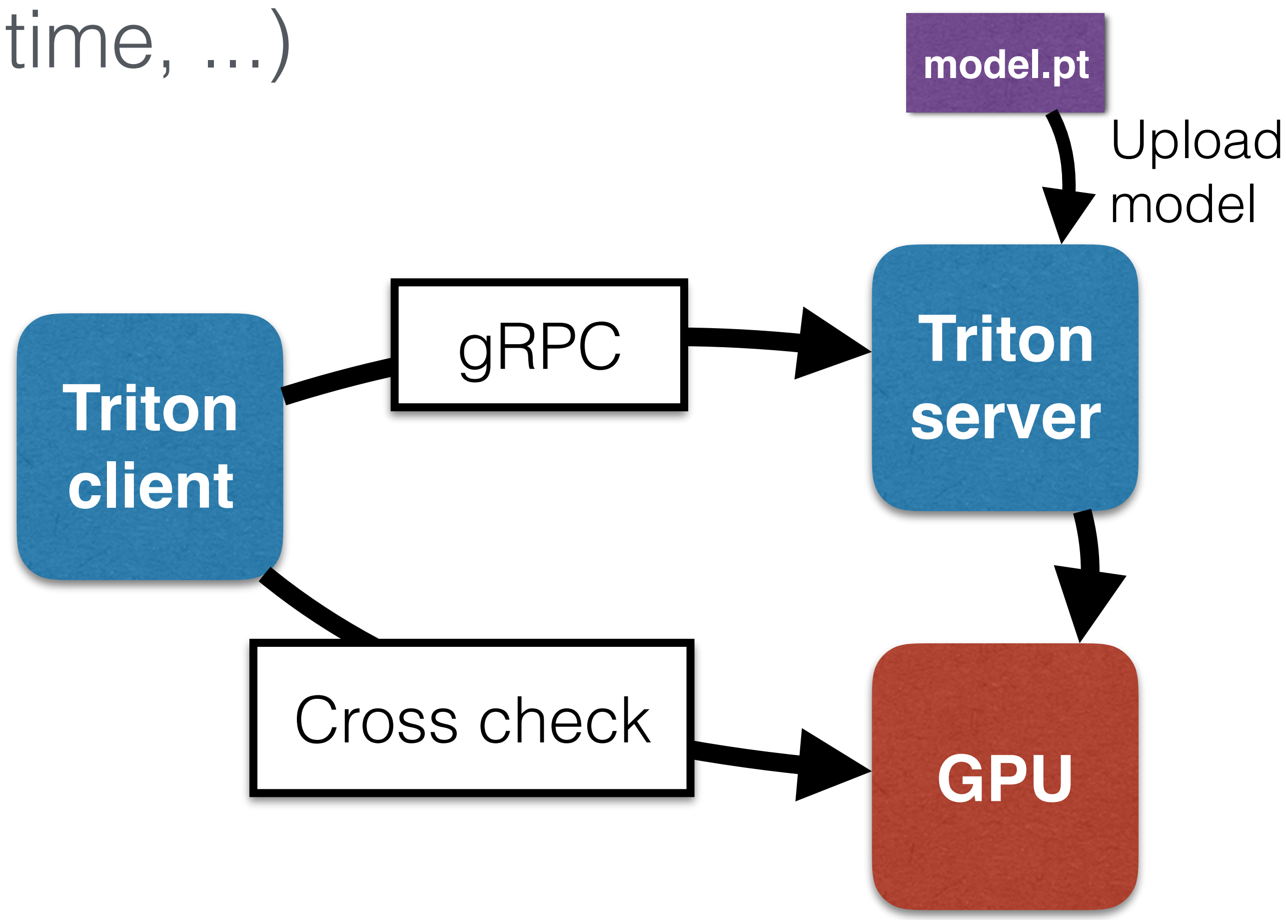
- Single particle performance pretty good
- Instance segmentation for a many-particle event still in active development by Alex Schuy (UW) et al.

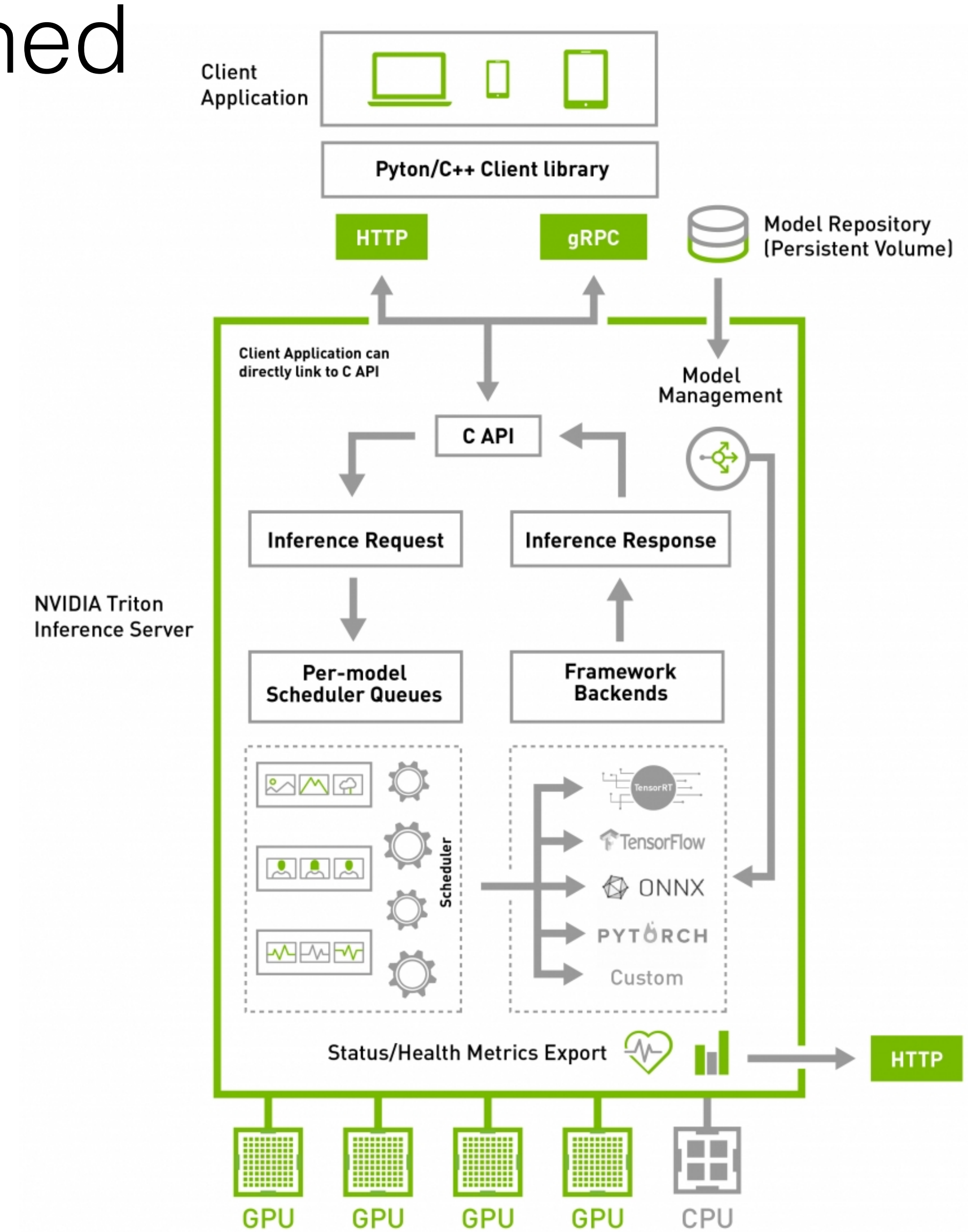
Accelerated ML: Nvidia Triton

- **Co-processors:**



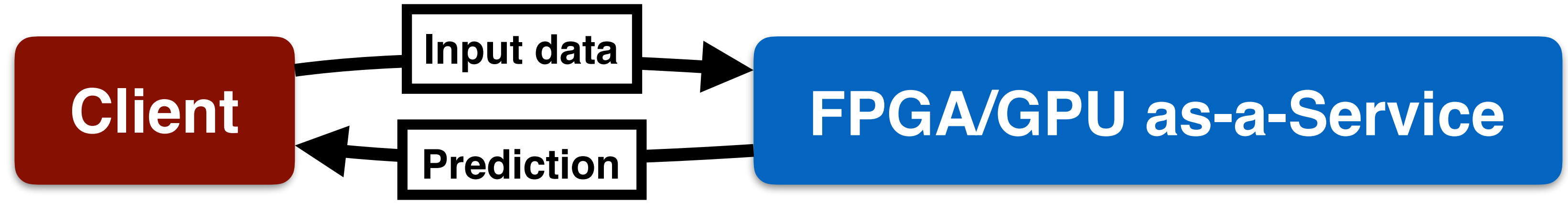
- Open source **inference serving software** that lets teams deploy trained AI models from (m)any framework(s) (TensorFlow, TensorRT, PyTorch, ONNX Runtime, ...)



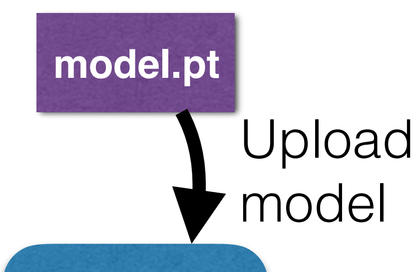


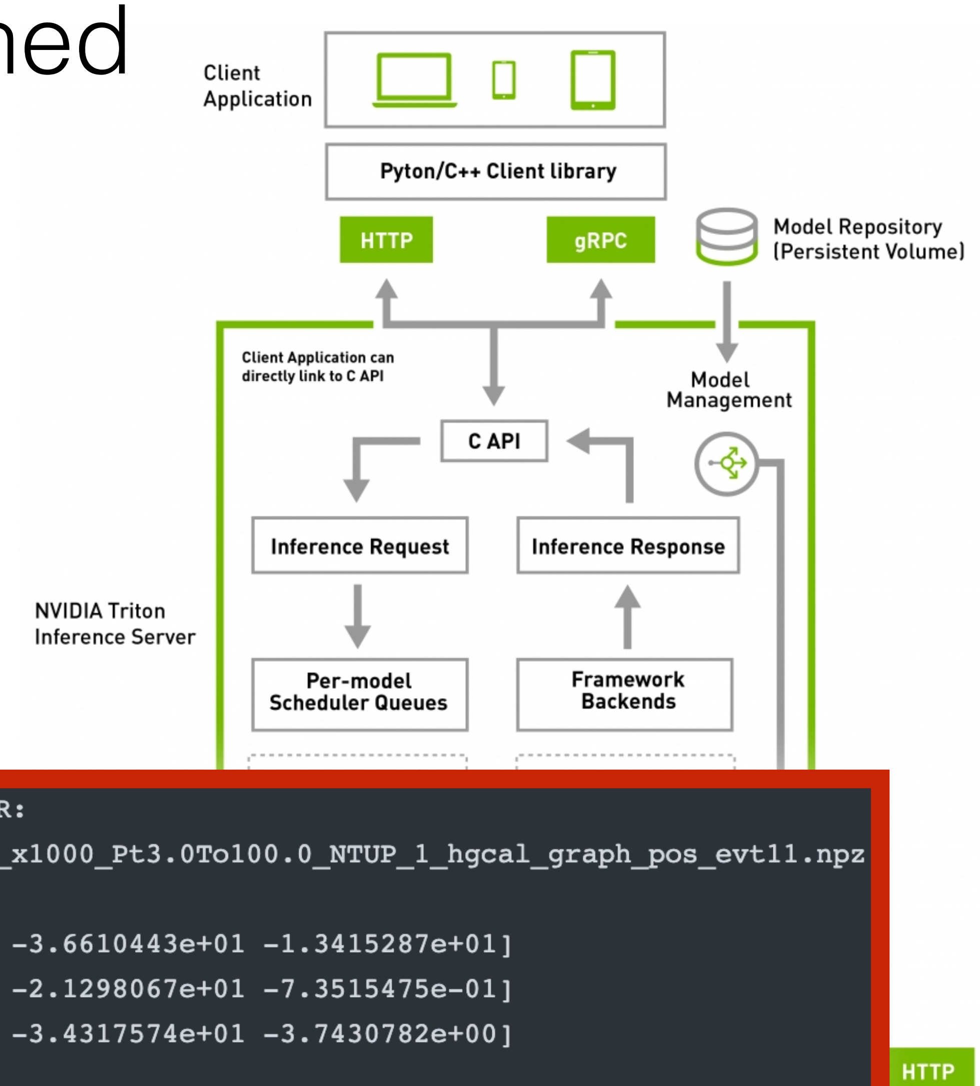
Accelerated ML: Nvidia Triton

- **Co-processors:**



- Open source **inference serving software** that lets teams deploy trained AI models from (m)any framework(s) (TensorFlow, TensorRT, PyTorch, ONNX Runtime, ...)






```
USING NON JIT / REGULAR MODEL:
hgcal_testdata/partGun_PDGid15_x1000_Pt3.0To100.0_NTUP_1_hgcal_graph_pos_evt11.npz
(19610, 5) (2, 556450)
tensor([[-1.4305e-06, -1.7474e+01, -3.6610e+01, -1.3415e+01],
        [-6.5627e-01, -6.3304e+00, -2.1298e+01, -7.3515e-01],
        [-2.3984e-02, -1.0954e+01, -3.4318e+01, -3.7431e+00],
        ...,
        [-1.0625e+01, -7.8196e+00, -1.5301e+01, -4.2653e-04],
        [-1.0433e+01, -7.5542e+00, -1.4863e+01, -5.5385e-04],
        [-1.0380e+01, -7.5484e+00, -1.4726e+01, -5.5861e-04]],
```

```
USING JIT MODEL ON TRITON SERVER:
/hgcal_testdata/partGun_PDGid15_x1000_Pt3.0To100.0_NTUP_1_hgcal_graph_pos_evt11.npz
(19610, 5) (2, 556450)
[[-1.4305115e-06 -1.7474344e+01 -3.6610443e+01 -1.3415287e+01]
 [-6.5626115e-01 -6.3303695e+00 -2.1298067e+01 -7.3515475e-01]
 [-2.3983955e-02 -1.0954206e+01 -3.4317574e+01 -3.7430782e+00]
 ...
 [-1.0624674e+01 -7.8196464e+00 -1.5300532e+01 -4.2653084e-04]
 [-1.0433395e+01 -7.5541725e+00 -1.4862620e+01 -5.5384636e-04]
 [-1.0380036e+01 -7.5483656e+00 -1.4725585e+01 -5.5861473e-04]]
```

Now works for our HGCAL clustering network

Other GPU inference results in recent paper submission: https://arxiv.org/abs/2007.10359

Conclusion

- ML reconstruction in calorimeters looks promising
  - Clustering is the hardest problem
    - Multiple valid architectures are available, early good results for GNN approaches
  - Have not yet tried ML on datasets with PU, so final conclusions cannot be made yet
  - Planning to improve our input simulation dataset with better truth definitions and PU
    - Hoping for new results by the summer

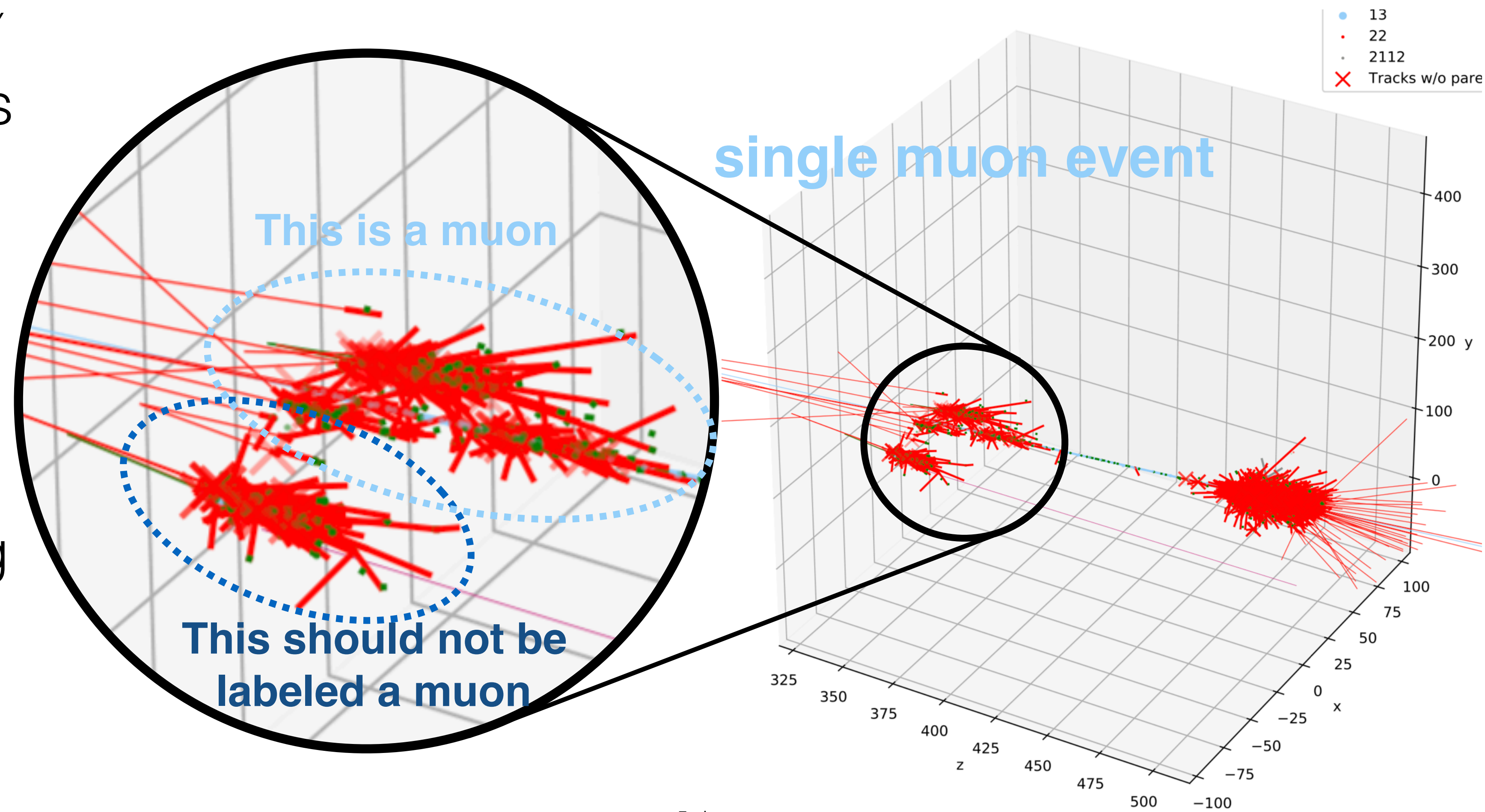
Backup

Fermilab

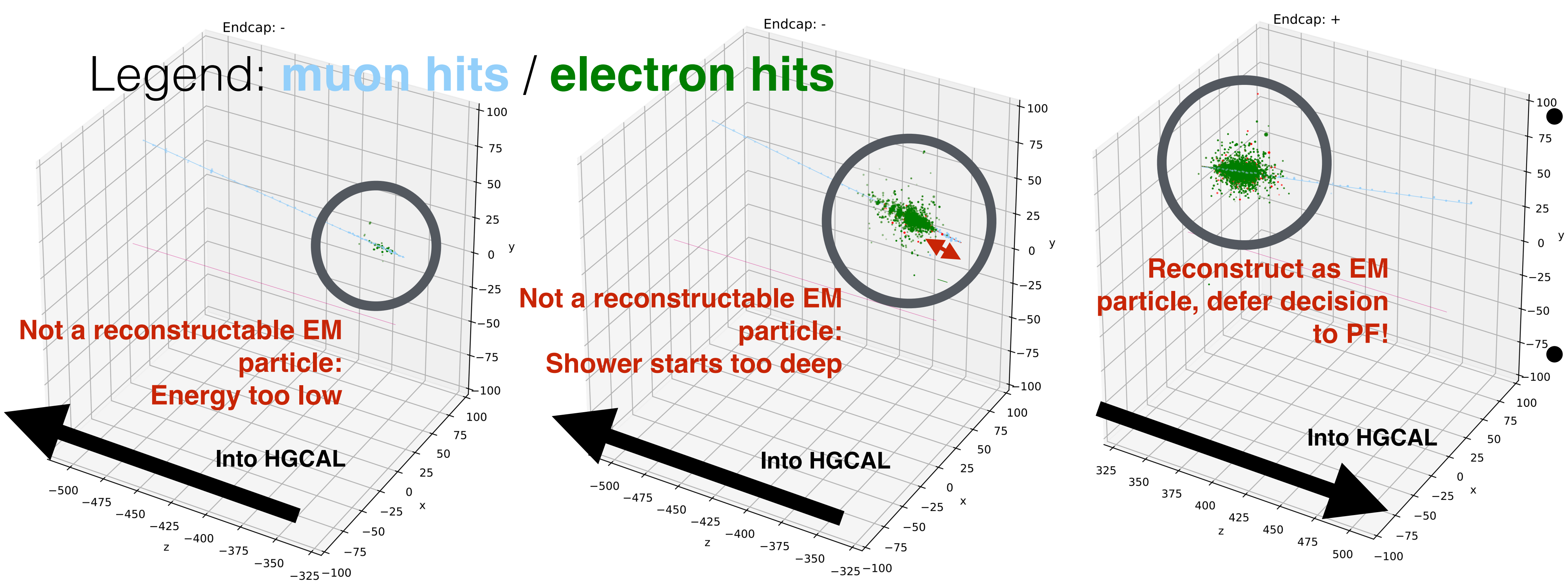
Truth in HGCAL

With great resolution comes great responsibility

- HGCAL able to resolve brems as reconstructable particles, but MC labels **secondary particles the same as their parent particle**
- Trying to define a better truth definition, sometimes labeling secondary particles as their own reconstructable particle



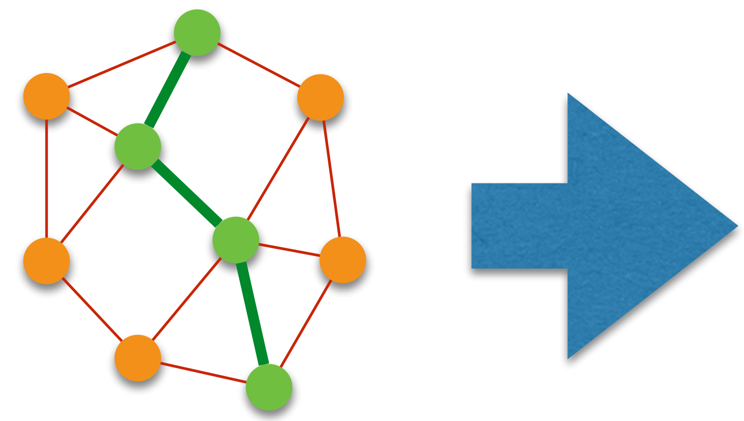
Legend: muon hits / electron hits



- No clear figure of merit
- Technically hard (Geant/CMSSW)

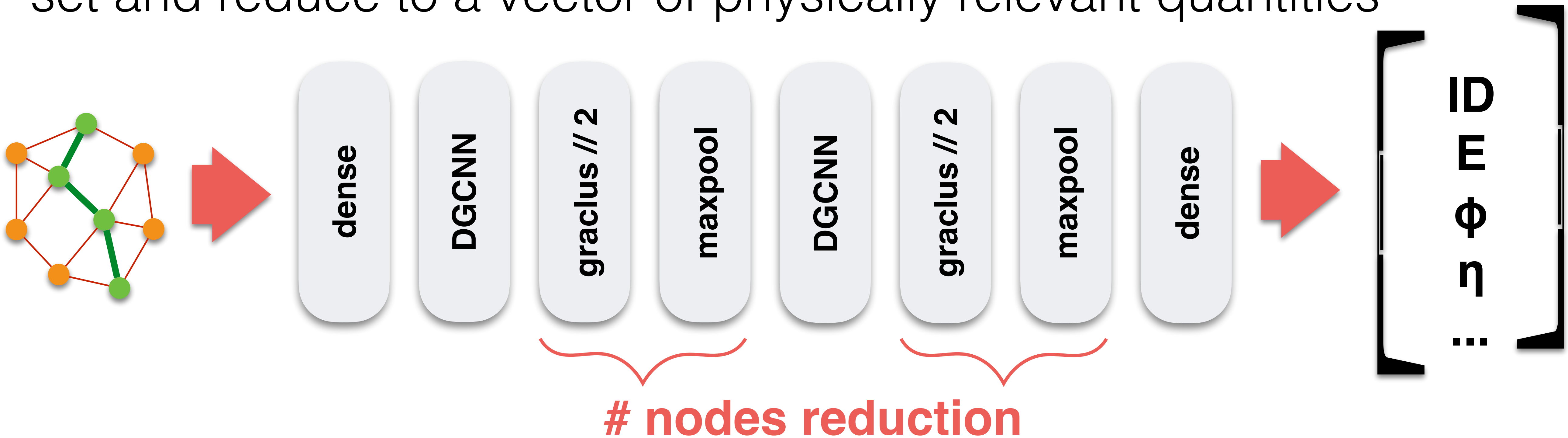
DRN: Property determination

- Primary task of GNN: **Clustering** , need to translate clusters to physics



ID, energy, 4-mom, ...

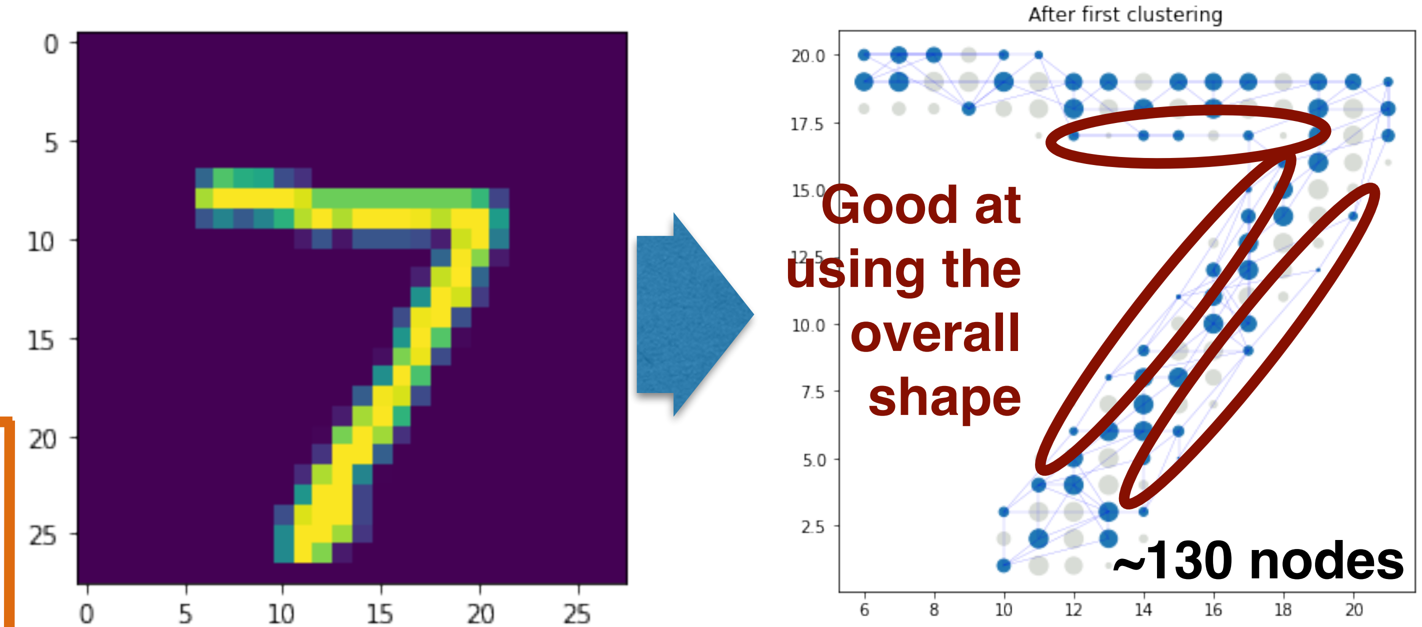
- '**Dynamic Reduction Network**' capable of taking an unordered set and reduce to a vector of physically relevant quantities

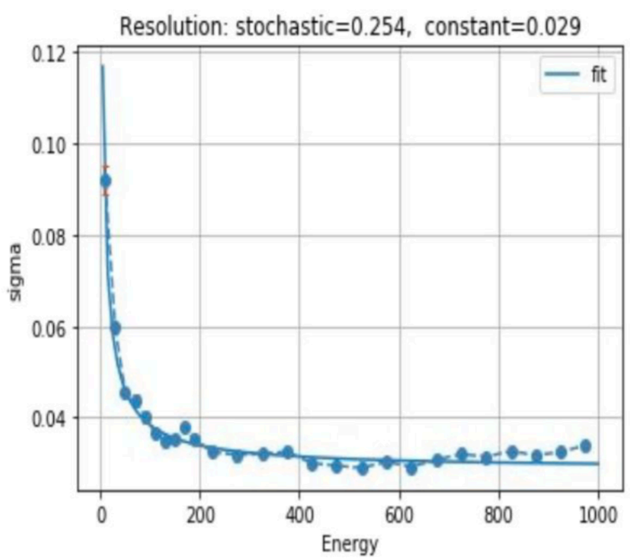


- Because of *EdgeConv*, **learns** how to use **organization and weighting** of input data to regress to physics

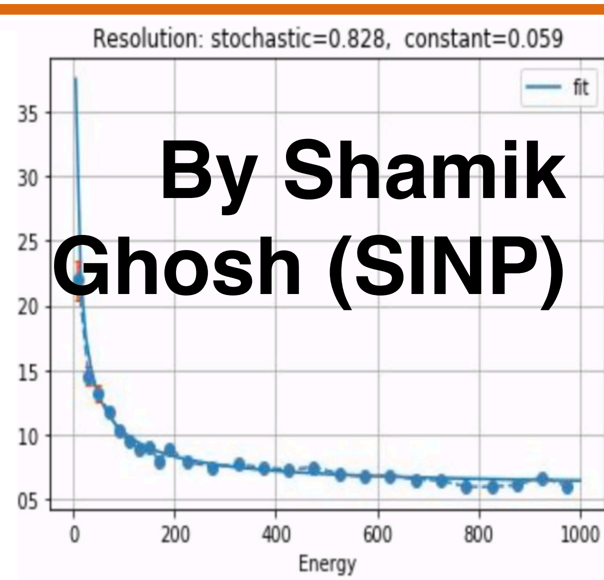
DRN in action

On **MNIST:** Mostly to enhance our understanding
Performs **~25th** in the **world ranking** for MNIST





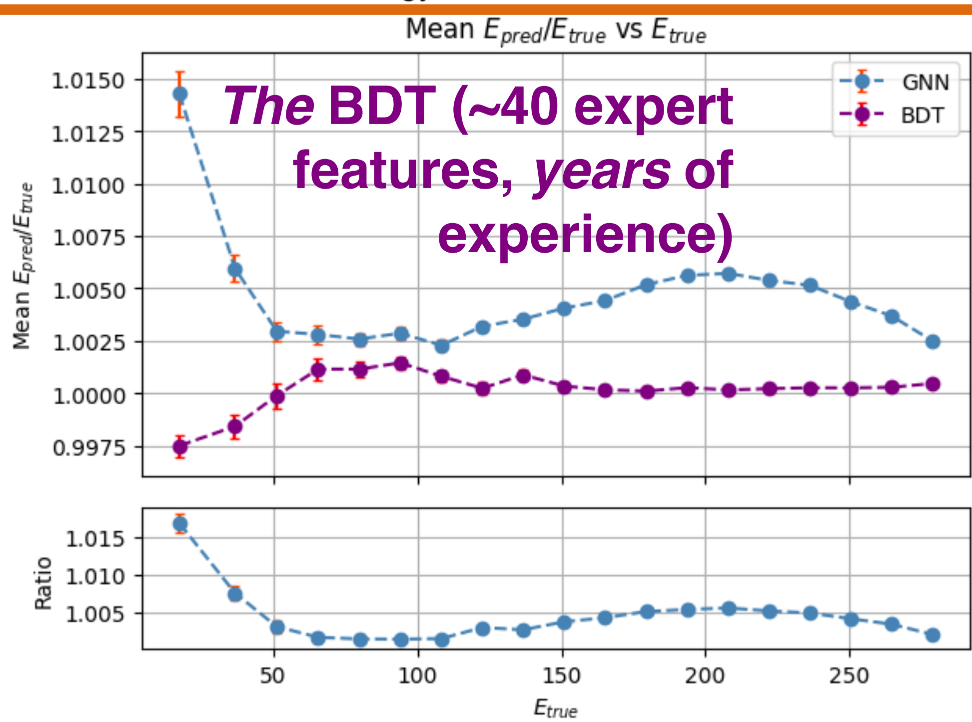
Photon energy resolution

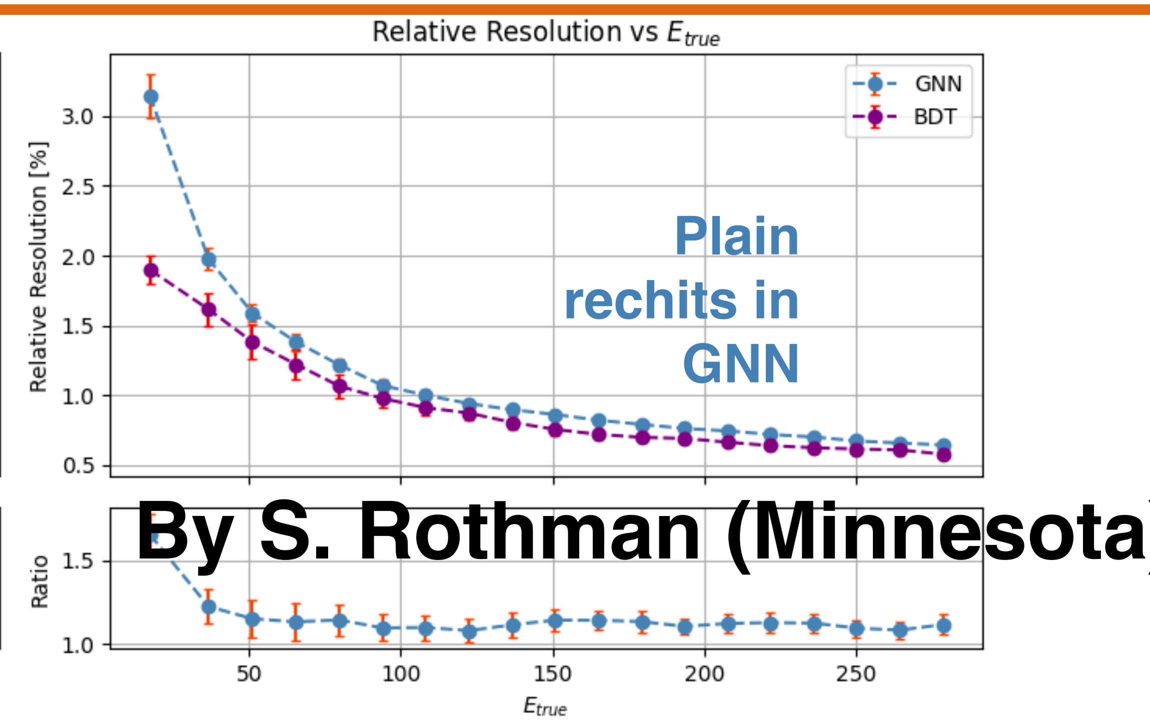


Pion energy resolution

HGCAL energy regression for photons and pions in pretty good shape

Performs very well for energy regression in **ECAL**





DRN interpretation: MNIST

- Trying out DRN on well-studied problem
- **MNIST:** Hand-written character recognition
- probably the only NN not using the whitespace → memory efficiently

(k=4 inside DGCNN)

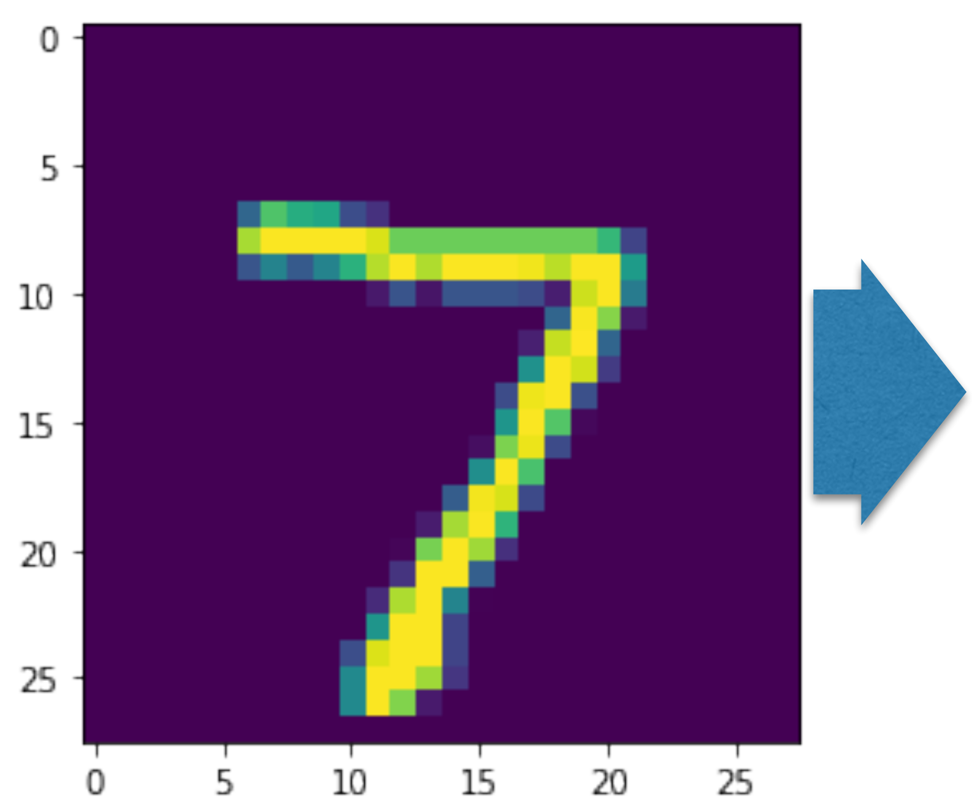
HD	Test acc
256	0.9955
128	0.9947
64	0.9930
32	0.9899
20	0.9854

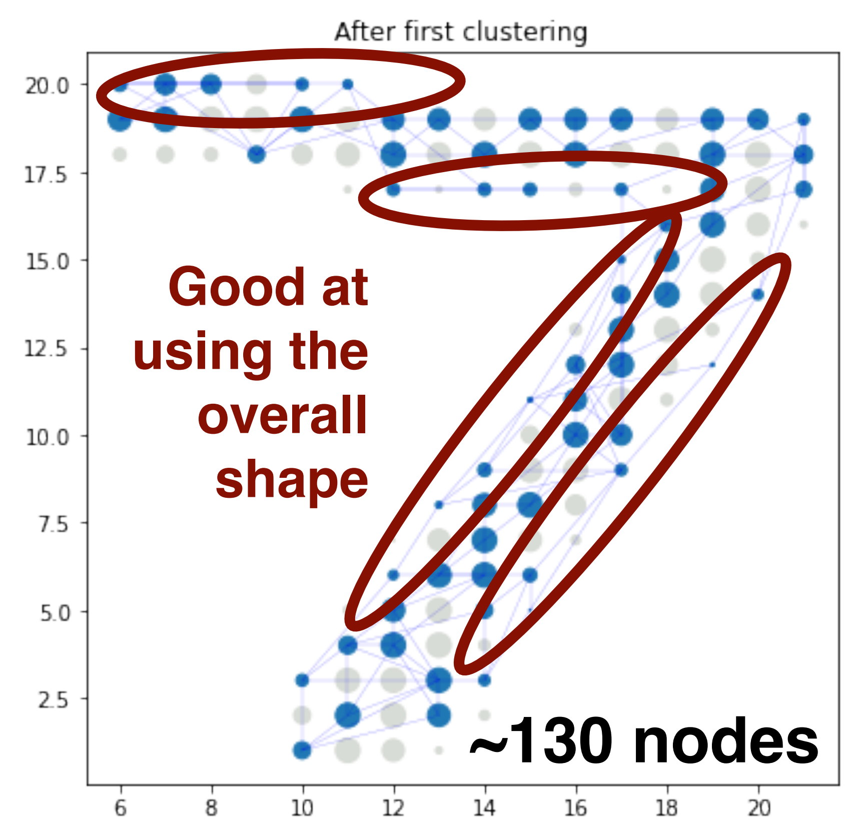
MNIST Leaderboard

https://paperswithcode.com/sota/image-classification-on-mnist

DRN somewhere here

		% error rate (1-acc)
24	MLR DNN ()	0.4
25	MIM	0.4
26	Fitnet-LSUV-SVM	0.4
27	Deformation Models ()	0.5
28	Trainable feature extractor ()	0.5
29	The Best Multi-Stage Architecture ()	0.5
30	COSFIRE ()	0.5
31	Maxout Networks	0.5

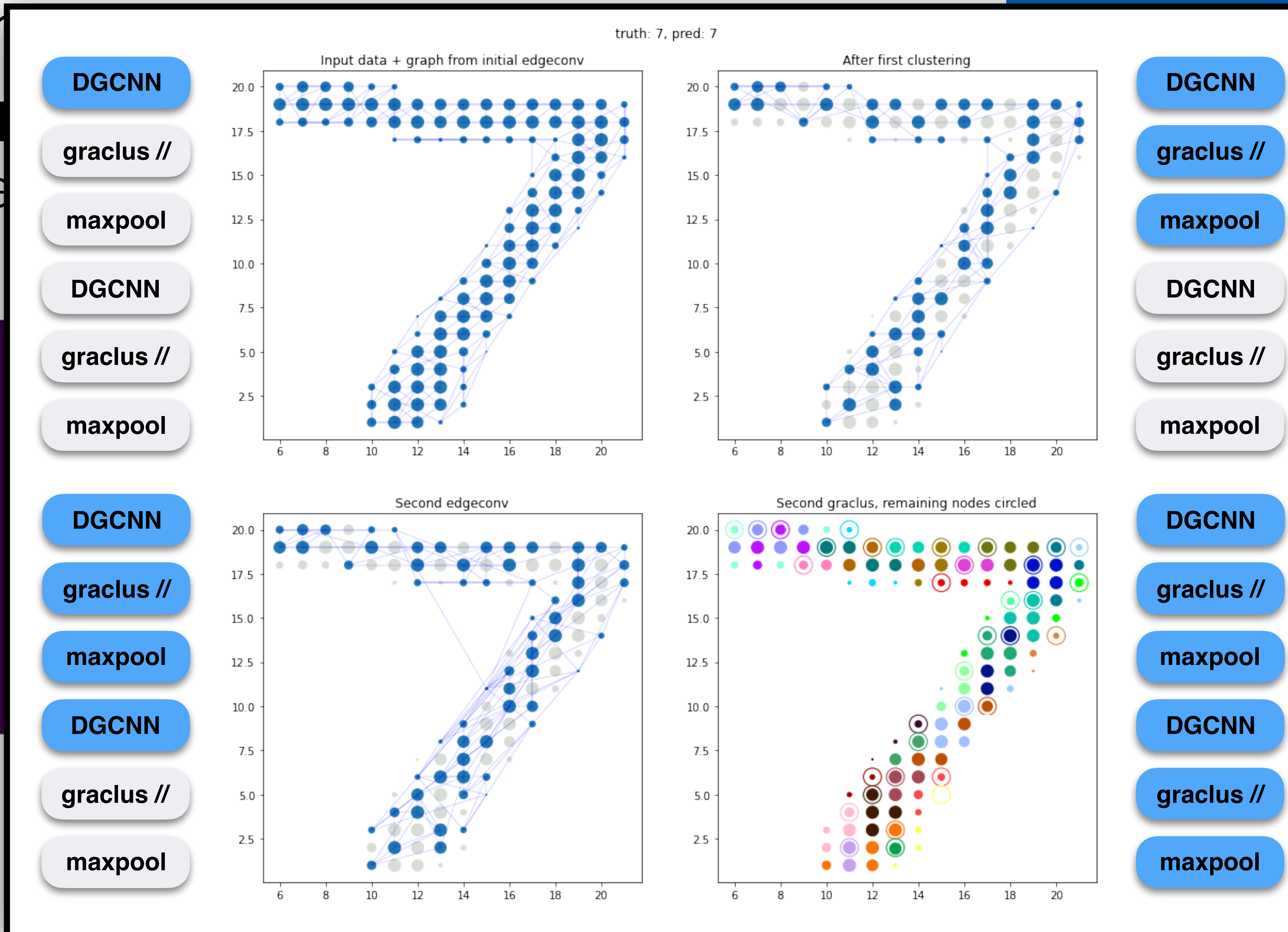




Interpretation: MNIST

(k=4 inside DGCNN)

- Tr
- **M** re



st acc	
9955	
9947	
9930	
9899	
9854	

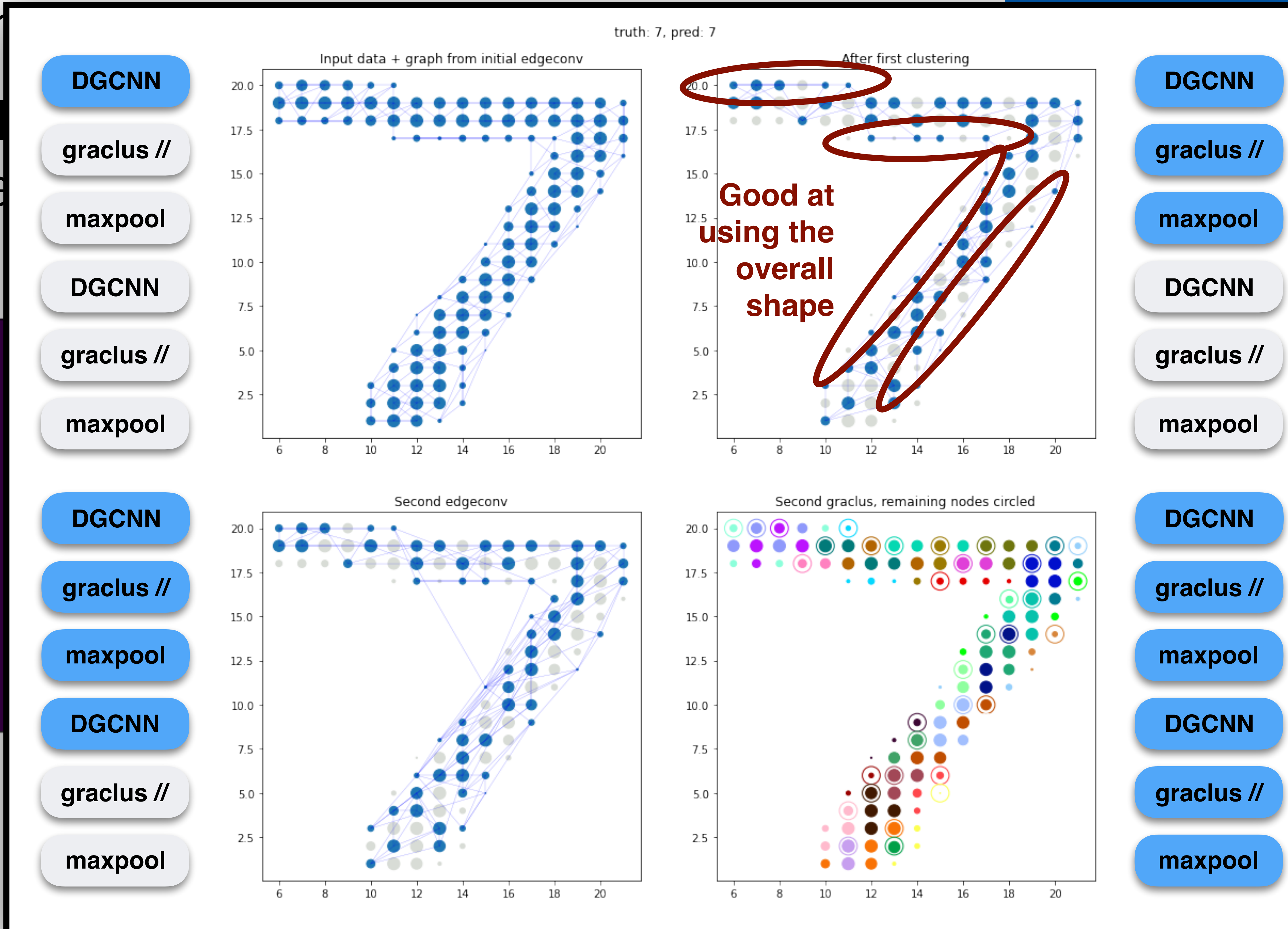
oard

ssification-on-mnist

	0.4
	0.4
	0.4
	0.5
or	0.5
	0.5
	0.5
	0.5

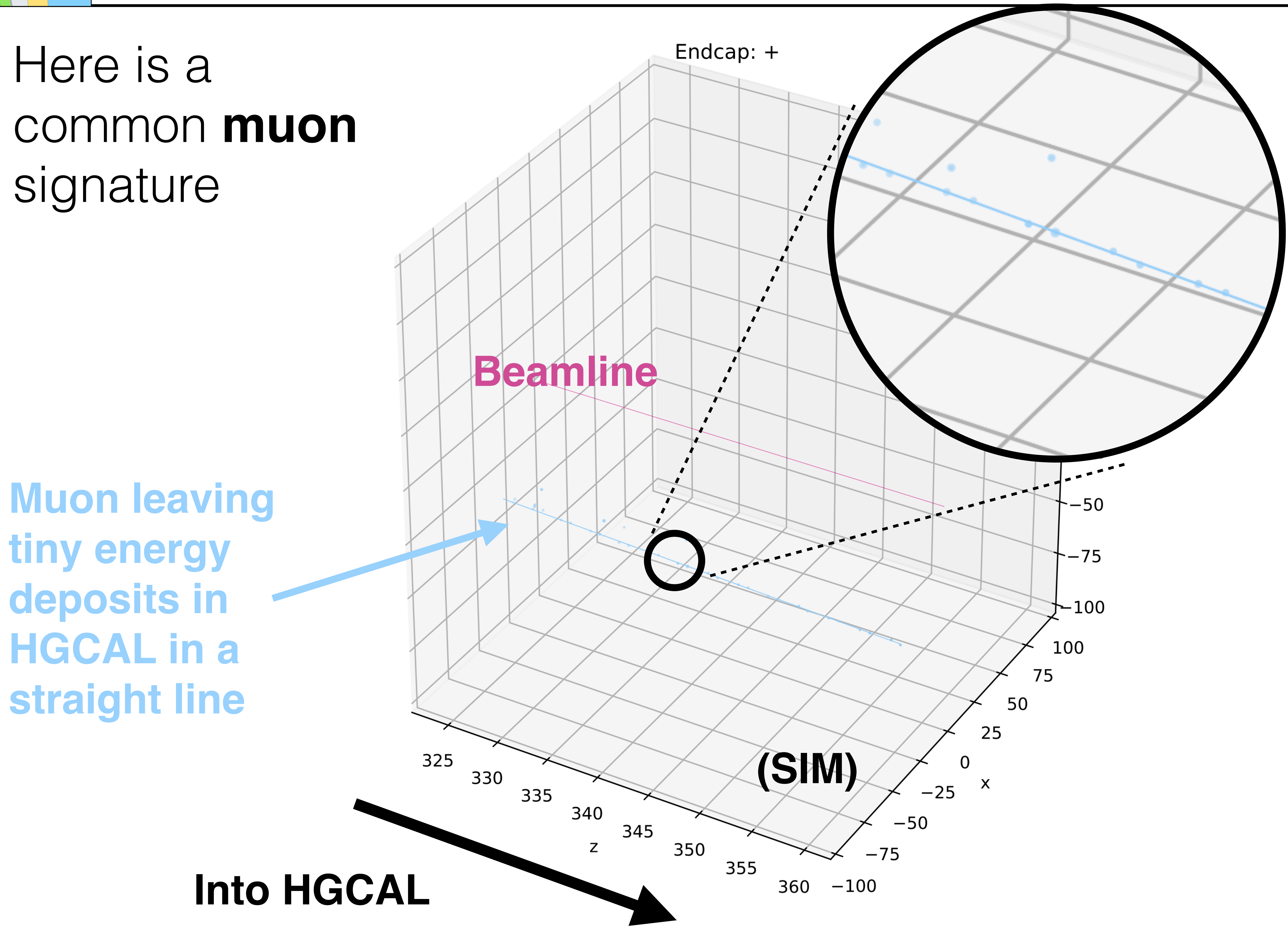
Interpretation: MNIST

(k=4 inside DGCNN)



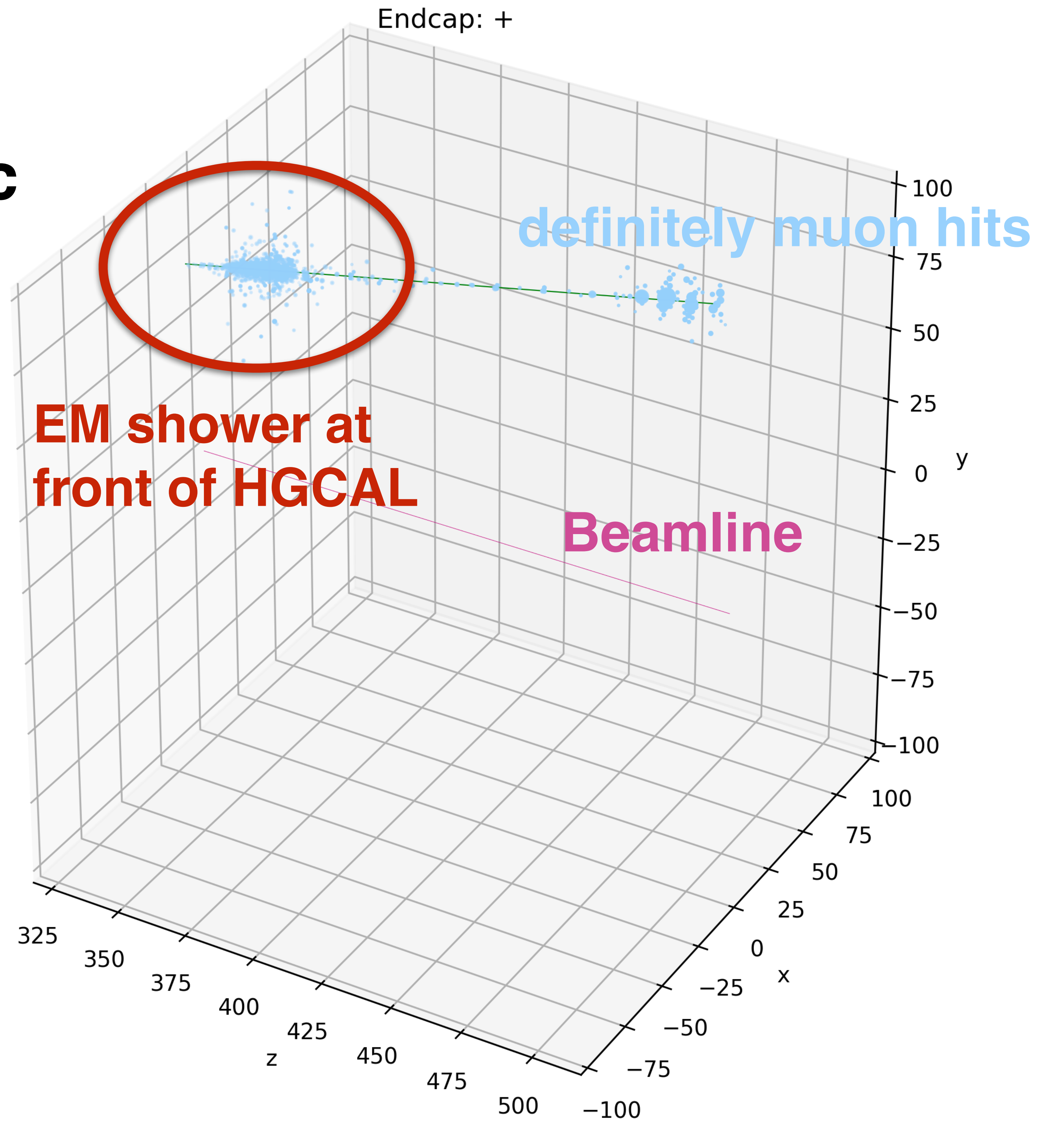
Truth in HGCAL

Here is a common **muon** signature



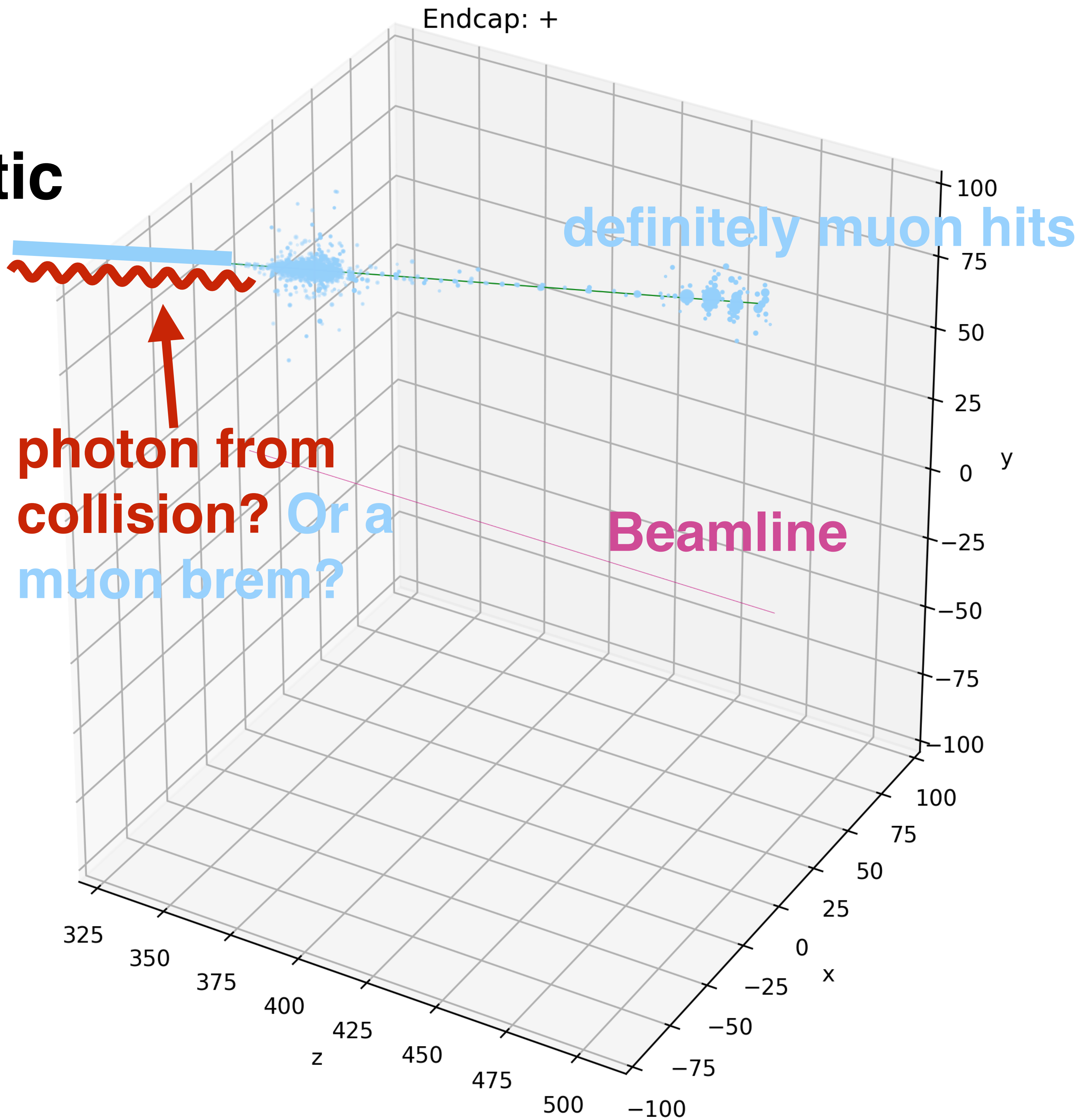
The problem

Sometimes a μ **radiates** an **electromagnetic (EM) particle**



The problem

Sometimes a μ **radiates** an **electromagnetic (EM) particle**

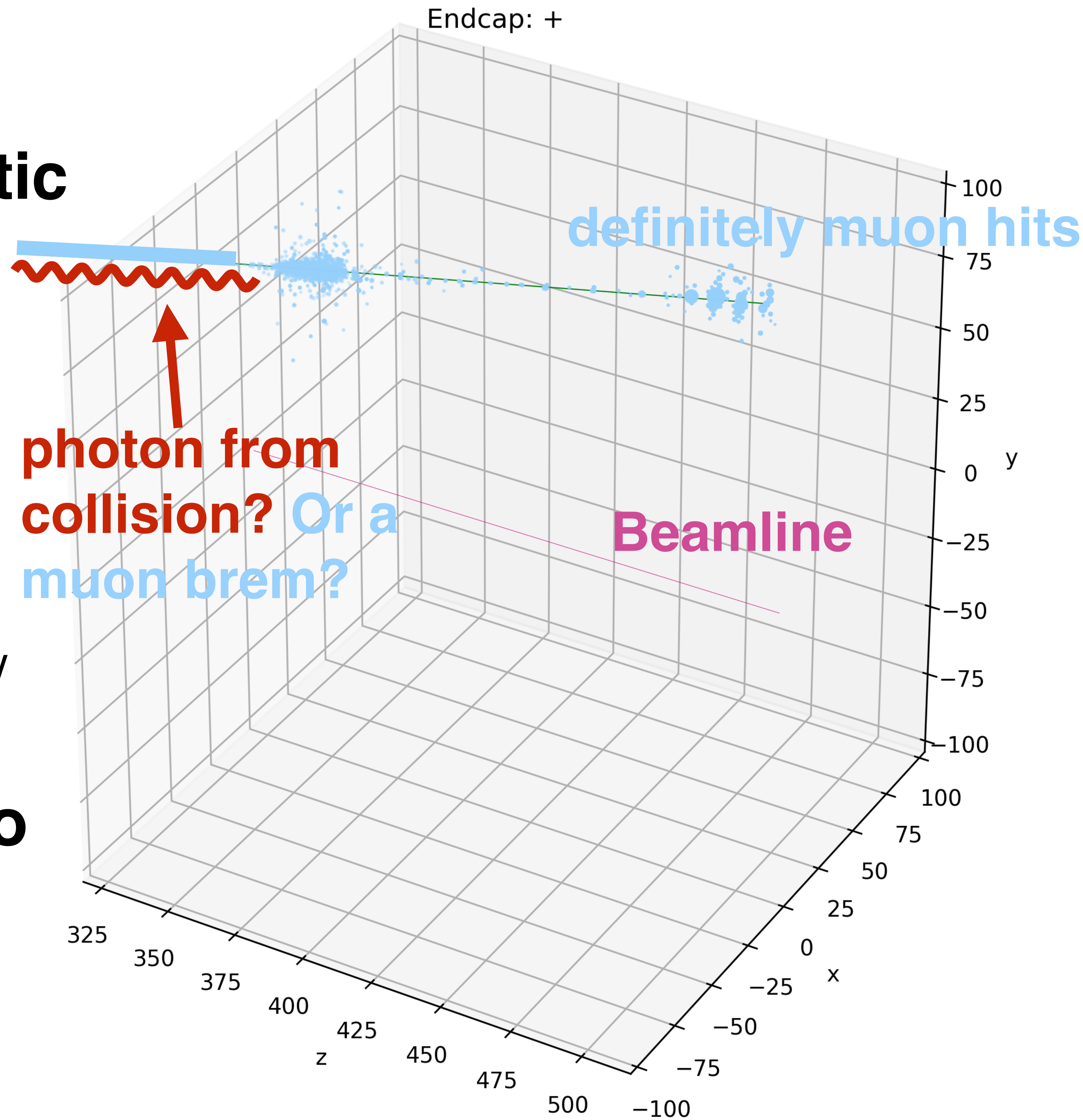


The problem

Sometimes a μ **radiates** an **electromagnetic (EM) particle**

This is a pure muon simulation, so it's a brem now

But we want to reconstruct this as a separate photon

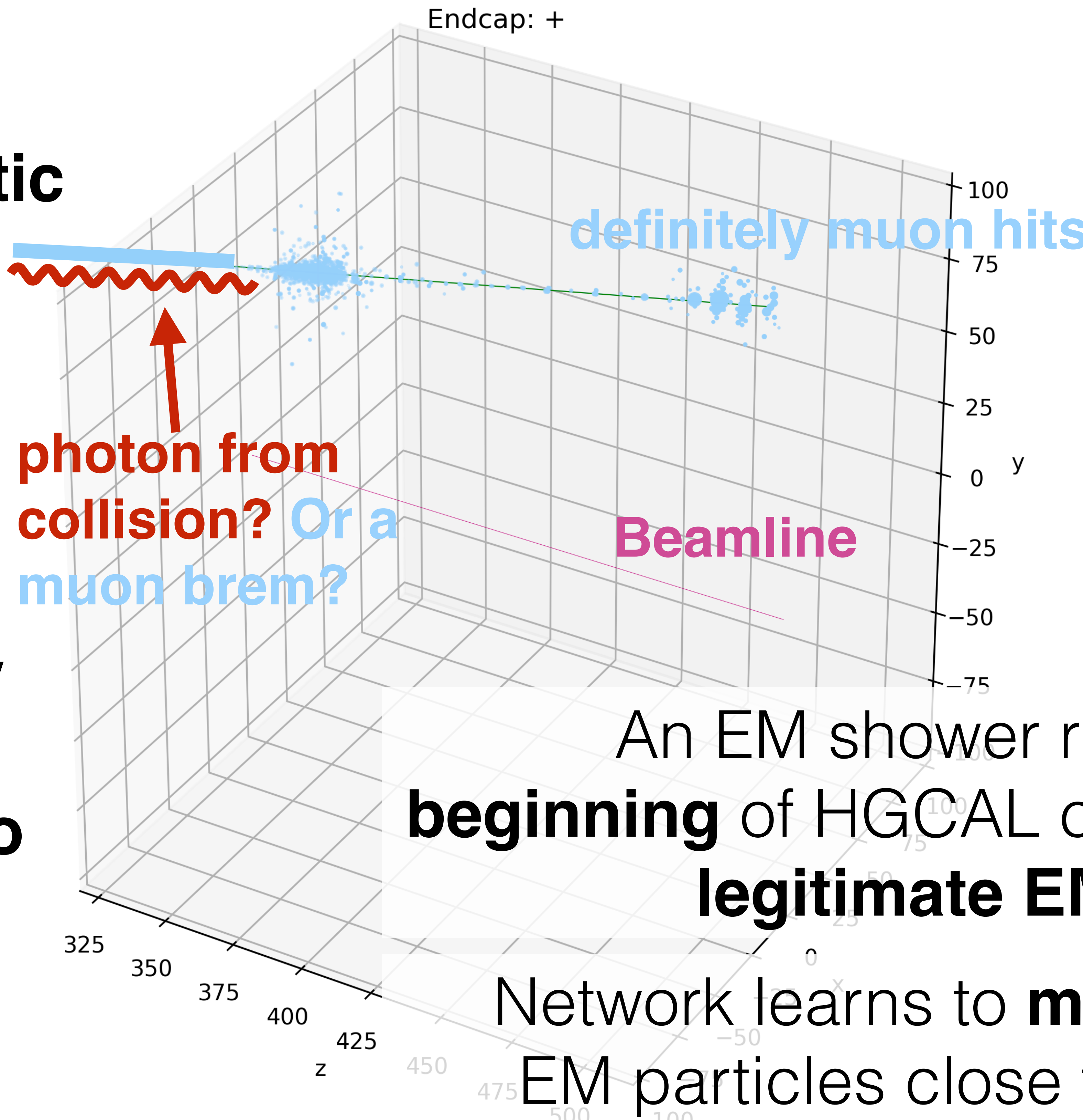


The problem

Sometimes a μ **radiates** an **electromagnetic (EM) particle**

This is a pure muon simulation, so it's a brem now

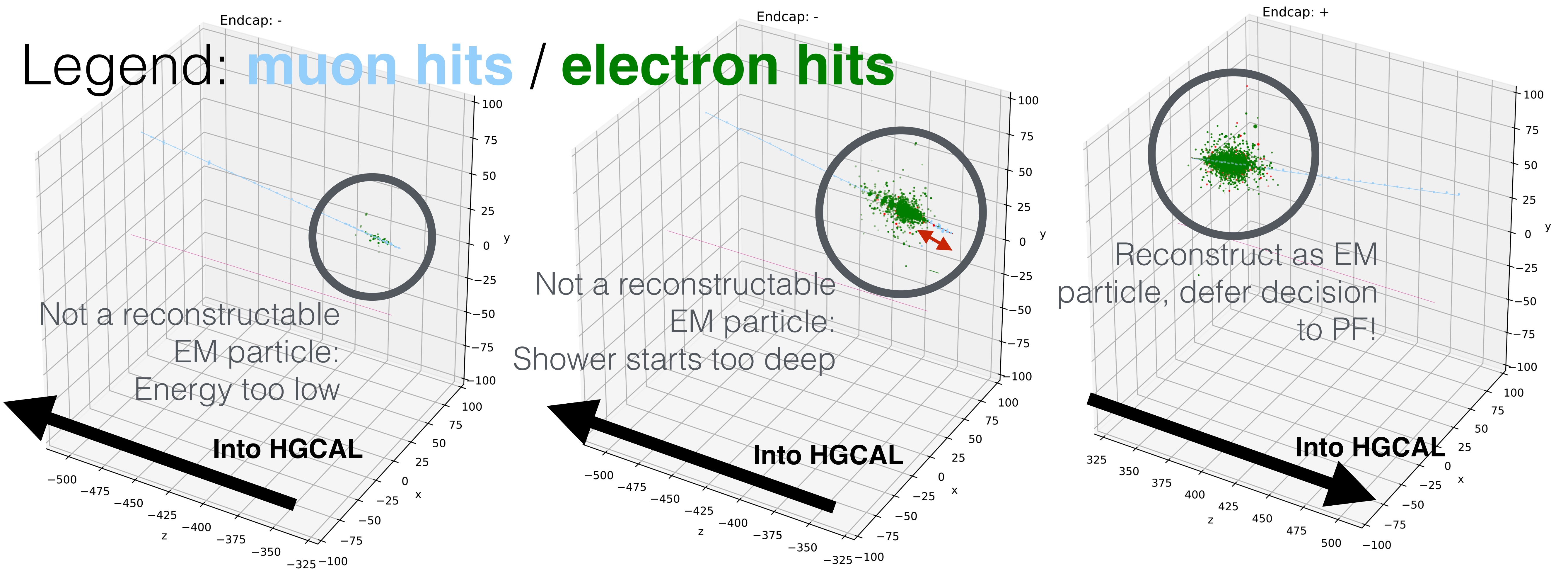
But we want to reconstruct this as a separate photon



An EM shower right at the **beginning** of HGCAL could be a **legitimate EM particle**

Network learns to **misidentify** EM particles close to muons!

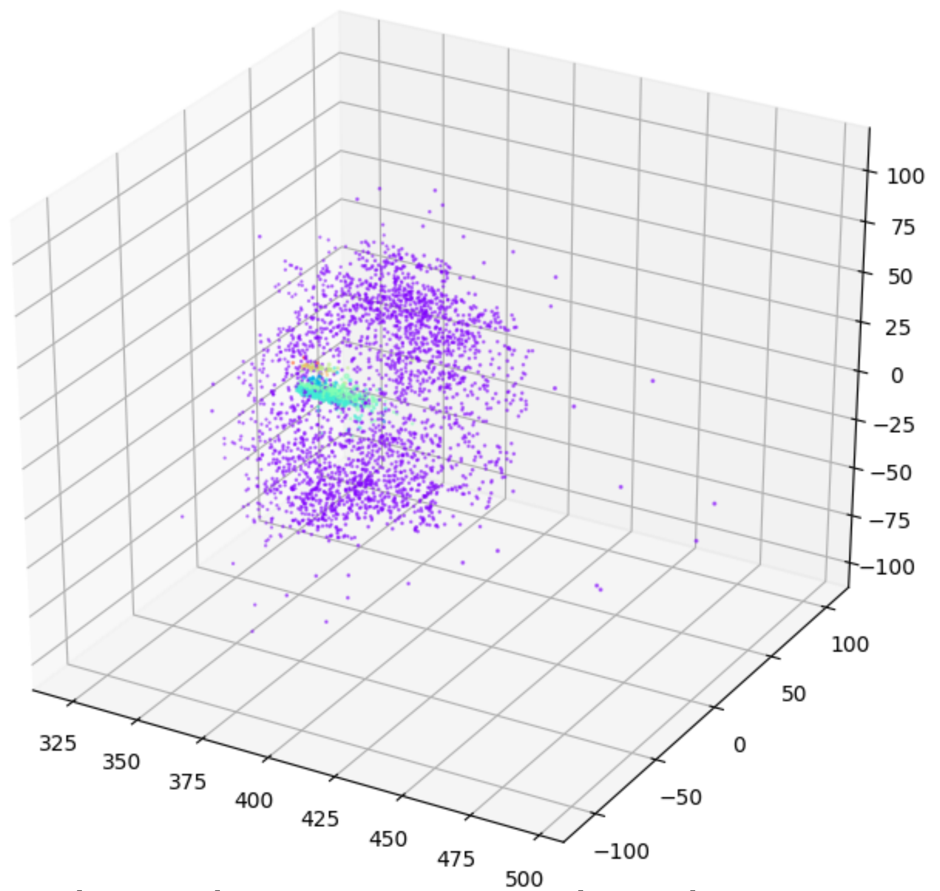
Work in progress

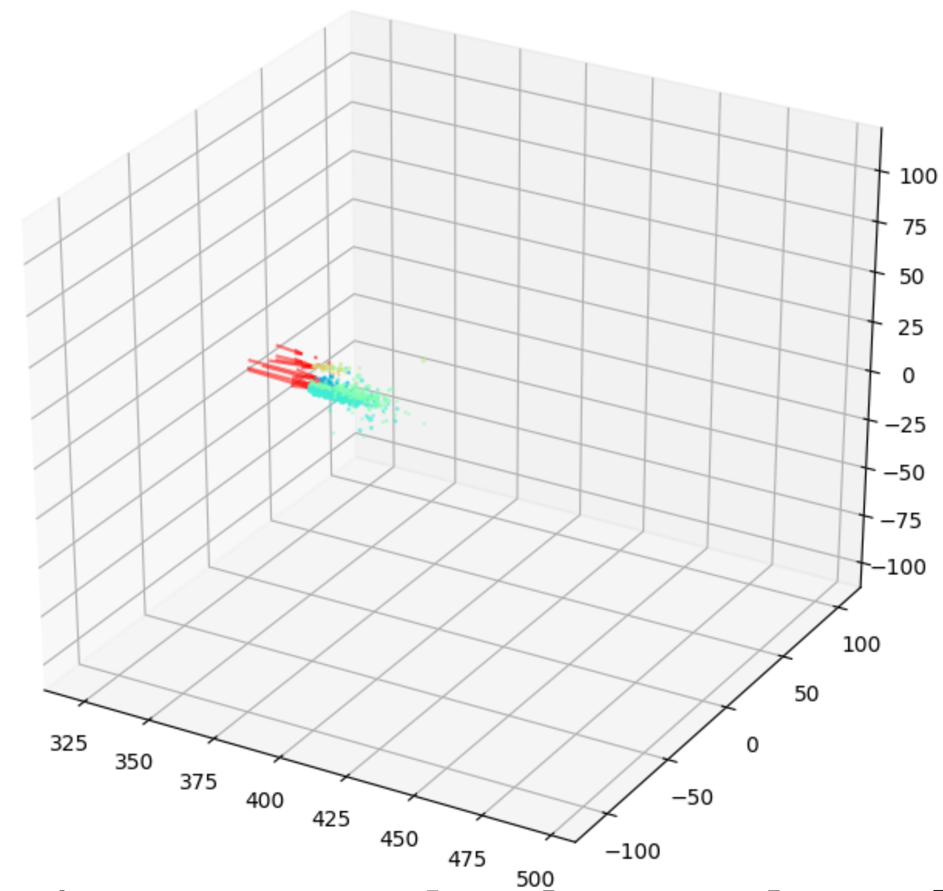


- Criterion for a 'reconstructable EM particle' is **soft**
- Track history bookkeeping by Geant as implemented in CMSSW is obscure
- Starting **CMSSW 11_1_0_pre6**, able to **assign hits to child tracks** with a configurable energy threshold
- Mis-reconstruction of 'real' photons (from the collision) as brem photons (or vice versa) has all sort of effects on **PF**
    - Problem of truth definition is broader than just HGCAL

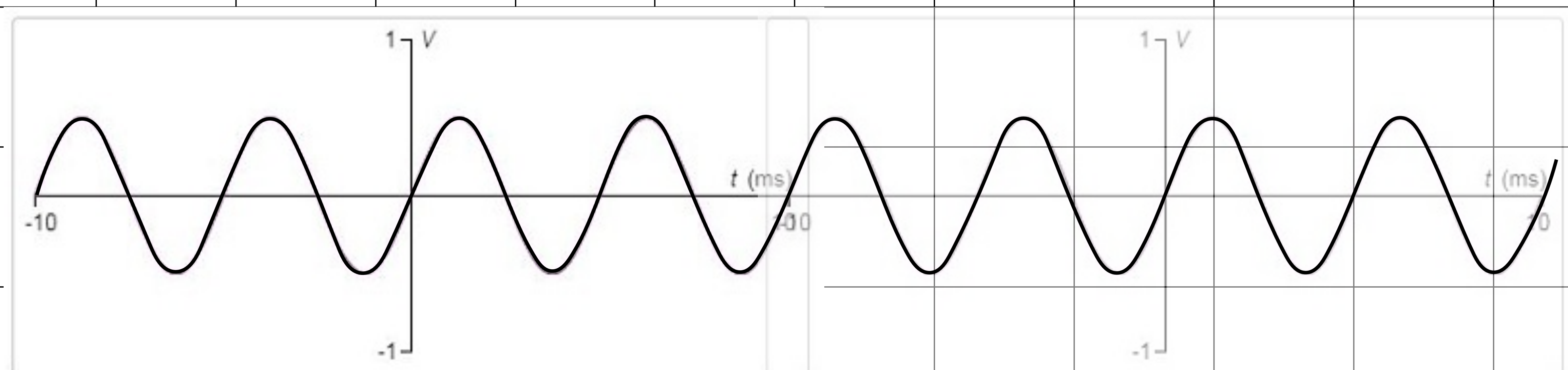
Boundary crossing criterion

- More broadly in default CMSSW: **We do not know exactly what the track 4-vector is for each shower**
  - Hits only point to their parent particle, not to secondary particles from the SIM step





- Trace back to see whether (secondary) tracks **cross the boundary into HGCAL**, *and* are **sufficiently energetic**
  - Possible to define the "true" particle ID and cluster as "the object that entered HGCAL"


1
V
t (ms)
-10
-1
1
V
t (ms)
-1